

23-1428

**In The
United States Court Of Appeals
For The Federal Circuit**

**EGERA, INC.,
*Plaintiff-Appellant***

v.

**CISCO SYSTEMS, INC.,
*Defendant-Appellee***

**Appeal from the United States District Court for the District of Massachusetts
in Case No. 1:16-cv-11613-RGS, Judge Richard G. Stearns**

OPENING BRIEF OF PLAINTIFF-APPELLANT

**Robert R. Brunelli
Matthew C. Holohan
SHERIDAN ROSS P.C.
1560 Broadway, Suite 1200
Denver, Colorado 80202
Telephone: (303) 863-9700
Facsimile: (303) 863-0223**

Counsel for Plaintiff-Appellant

Dated: April 26, 2023

REPRESENTATIVE CLAIMS AT ISSUE

5. A method of automatically deploying at least one virtual processing area network, in response to software commands, said method comprising the acts of:

providing a platform having a plurality of computer processors and at least one control node connected to an internal communication network, wherein the at least one control node is in communication with an external communication network and an external storage network having an external storage address space;

receiving software commands specifying (i) a number of processors for a virtual processing area network, (ii) a virtual local area network topology defining interconnectivity and switching functionality among the specified processors of the virtual processing area network, and (iii) virtual storage space for the virtual processing area network;

under programmatic control and in response to the software commands, selecting a corresponding set of computer processors for the virtual processing area network;

under programmatic control and in response to the software commands, programming said corresponding set of computer processor; and the internal communication network to establish the specified virtual local area network topology providing communication among said corresponding set of computer processors but excluding the processors from the plurality not in said set;

under programmatic control and in response to the software commands, programming the at least one control node to define a virtual storage space of the virtual processing network, said virtual storage space having a defined correspondence to a subset of the external storage address space of the external storage network;

wherein messages from the plurality of computer processors to the external communication network and to the external storage network are received and modified by the least one control node which transmits the modified messages to the external communication network and to the external storage network; and

wherein the plurality of computer processors and the at least one control node emulate Ethernet functionality over the internal communication network.

7. A method of automatically deploying at least one virtual processing area network, in response to software commands, said method comprising the acts of:

providing a platform having a plurality of computer processors and at least one control node connected to an internal communication network, wherein the at least one control node is in communication with an external communication network and an external storage network having an external storage address space;

receiving software commands specifying (i) a number of processors for a virtual processing area network, (ii) a virtual local area network topology defining interconnectivity and switching functionality among the specified processors of the virtual processing area network, and (iii) virtual storage space for the virtual processing area network;

under programmatic control and in response to the software commands, selecting a corresponding set of computer processors for the virtual processing area network;

under programmatic control and in response to the software commands, programming said corresponding set of computer processor; and the internal communication network to establish the specified virtual local area network topology providing communication among said corresponding set of computer processors but excluding the processors from the plurality not in said set;

under programmatic control and in response to the software commands, programming the at least one control node to define a virtual storage space of the virtual processing network, said virtual storage space having a defined correspondence to a subset of the external storage address space of the external storage network;

wherein messages from the plurality of computer processors to the external communication network and to the external storage network are received and modified by the least one control node, which transmits the modified messages to the external communication network and to the external storage network;

wherein the at least one control node receives, via the internal communication network, storage messages from said corresponding set of computer processors, axed wherein the at least one control node extracts am address from a received storage message, identifies the defined corresponding address in the external storage address space, and provides messages on the external storage network corresponding to the received storage messages and having the corresponding address; and

wherein the at least one control node buffers data corresponding to write messages received from a computer processor of said corresponding set of computer processors and provides the buffered data in the corresponding message provided to the external storage network.

**UNITED STATES COURT OF APPEALS
FOR THE FEDERAL CIRCUIT****CERTIFICATE OF INTEREST****Case Number** 23-1428**Short Case Caption** Egenera, Inc. v. Cisco Systems, Inc.**Filing Party/Entity** Egenera, Inc.**Instructions:**

1. Complete each section of the form and select none or N/A if appropriate.
2. Please enter only one item per box; attach additional pages as needed, and check the box to indicate such pages are attached.
3. In answering Sections 2 and 3, be specific as to which represented entities the answers apply; lack of specificity may result in non-compliance.
4. Please do not duplicate entries within Section 5.
5. Counsel must file an amended Certificate of Interest within seven days after any information on this form changes. Fed. Cir. R. 47.4(c).

I certify the following information and any attached sheets are accurate and complete to the best of my knowledge.

Date: 04/26/2023Signature: /s/ Matthew C. HolohanName: Matthew C. Holohan

1. Represented Entities. Fed. Cir. R. 47.4(a)(1).	2. Real Party in Interest. Fed. Cir. R. 47.4(a)(2).	3. Parent Corporations and Stockholders. Fed. Cir. R. 47.4(a)(3).
Provide the full names of all entities represented by undersigned counsel in this case.	Provide the full names of all real parties in interest for the entities. Do not list the real parties if they are the same as the entities. <input checked="" type="checkbox"/> None/Not Applicable	Provide the full names of all parent corporations for the entities and all publicly held companies that own 10% or more stock in the entities. <input checked="" type="checkbox"/> None/Not Applicable
Egenera, Inc.		

☐ Additional pages attached

4. Legal Representatives. List all law firms, partners, and associates that (a) appeared for the entities in the originating court or agency or (b) are expected to appear in this court for the entities. Do not include those who have already entered an appearance in this court. Fed. Cir. R. 47.4(a)(4).

☐ None/Not Applicable ☒ Additional pages attached

Rachael Bacha Ropes & Gray LLP	James R. Batchelder Ropes & Gray LLP	Nicholas D. Bortz Ropes & Gray LLP
Samuel L. Brenner Ropes & Gray LLP	Kevin L. Burgess Ropes & Gray LLP	Keyna T. Chow Ropes & Gray LLP
Even Gourvitz Ropes & Gray LLP	Emma Notis-McConarty Ropes & Gray LLP	Steven Pepe Ropes & Gray LLP

5. Related Cases. Other than the originating case(s) for this case, are there related or prior cases that meet the criteria under Fed. Cir. R. 47.5(a)?

☒ Yes (file separate notice; see below) ☐ No ☐ N/A (amicus/movant)

If yes, concurrently file a separate Notice of Related Case Information that complies with Fed. Cir. R. 47.5(b). **Please do not duplicate information.** This separate Notice must only be filed with the first Certificate of Interest or, subsequently, if information changes during the pendency of the appeal. Fed. Cir. R. 47.5(b).

6. Organizational Victims and Bankruptcy Cases. Provide any information required under Fed. R. App. P. 26.1(b) (organizational victims in criminal cases) and 26.1(c) (bankruptcy case debtors and trustees). Fed. Cir. R. 47.4(a)(6).

☒ None/Not Applicable ☐ Additional pages attached

23-1428

Egenera, Inc. v. Cisco Systems, Inc.

Form 9 Certificate of Interest

Section 4 - Legal Representatives for Egenera, Inc., con't

Ropes & Gray LLP

David A. Serati

Scott S. Taylor

Andrew N. Thomases

McKool Smith, P.C.

Christopher T. Bovenkamp

John B. Campbell

Jordan Z. Carson

Kathy H. Li

Michael McKool, Jr.

Michael J. Miguel

James E. Quigley

Avery R. Williams

Kevin L. Burgess

Perkins Coie LLP

Dan L. Bagatell

Andrew T. Dufresne

Martin E. Gilmore

Murphy & King, PC

David L. Evans

Steven M. Veenema

TABLE OF CONTENTS

REPRESENTATIVE CLAIMS AT ISSUE.....	i
CERTIFICATE OF INTEREST	iv
TABLE OF AUTHORITIES	xi
REQUEST FOR ORAL ARGUMENT.....	xiv
STATEMENT OF RELATED CASES.....	xv
I. INTRODUCTION	1
II. JURISDICTIONAL STATEMENT	2
III. STATEMENT OF THE ISSUE PRESENTED FOR APPEAL.....	2
IV. STATEMENT OF THE CASE AND FACTS	3
A. Egenera and the '430 Patent	3
B. Claim Construction	5
C. Cisco's UCS and Noninfringement Arguments.....	6
1. Cisco's Summary Judgment Noninfringement Position	7
2. Cisco's Noninfringement Positions at Trial.....	8
3. The Conduct of Trial	18
D. Post-Trial Proceedings.....	28
V. SUMMARY OF ARGUMENT.....	29
VI. ARGUMENT.....	30
A. Standard of Review	30
B. The District Court Erred as a Matter of Law in Entering Summary Judgment of Noninfringement of Claims 1 and 5.	31

C.	The District Court Erred in Denying Egenera’s JMOL Motion.	35
1.	No Reasonable Jury Could Conclude that the UCS Does Not Program CPUs to Establish Network Topology.....	36
2.	No Reasonable Jury Could Conclude that the UCS Does Not Modify Messages to the External Communication Network.....	38
3.	No Reasonable Jury Could Conclude that the UCS Does Not Extract an Address and Identify a Corresponding Address for Messages to the External Storage Network.....	41
4.	The Jury’s Verdict Cannot Be Sustained Based on Other Claim Limitations.	45
5.	The Court Should Vacate the District Court’s Judgment and Remand for Further Proceedings.	46
D.	Egenera is Entitled to a New Trial Based on the Weight of the Evidence and the District Court’s Myriad Errors.	46
1.	Egenera’s Unrebutted Evidence of Infringement Warrants a New Trial.....	46
2.	The District Court’s Erroneous Jury Instruction Regarding “Copying” Warrants a New Trial.	47
3.	The District Court’s Refusal to Instruct the Jury that a Patented Product May Infringe Another Patent Warrants a New Trial.....	50
4.	Cisco’s Improper Closing Arguments Warrant a New Trial.....	52
5.	Cisco’s Improper Lay Opinion Testimony Warrants a New Trial.....	54
6.	The District Court Erred in Denying Egenera’s Motion for a New Trial.	55

VII. CONCLUSION AND STATEMENT OF RELIEF SOUGHT	58
ADDENDUM.....	60

TABLE OF AUTHORITIES

Cases

<i>AbbVie Deutschland GmbH & Co., KG v. Janssen Biotech, Inc.</i> , 759 F.3d 1285 (Fed. Cir. 2014)	30
<i>Allen Eng’g Corp. v. Bartell Indus., Inc.</i> , 299 F.3d 1336 (Fed. Cir. 2002)	20
<i>Anheuser-Busch, Inc. v. Nat. Beverage Distribs.</i> , 69 F.3d 337 (9th Cir. 1995)	53
<i>Astro-Med, Inc. v. Nihon Kohden Am., Inc.</i> , 591 F.3d 1 (1st Cir. 2009).....	31
<i>Chestnut v. City of Lowell</i> , 305 F.3d 18 (1st Cir. 2002).....	52
<i>Commil USA, LLC v. Cisco Sys., Inc.</i> , 720 F.3d 1361 (Fed. Cir. 2013), <i>vacated on other grounds</i> <i>Commil USA, LLC v. Cisco Sys., Inc.</i> , 135 S. Ct. 1920 (2018).....	54
<i>Crowe v. Bolduc</i> , 334 F.3d 124 (1st Cir. 2003).....	57
<i>Davignon v. Hodgson</i> , 524 F.3d 91 (1st Cir. 2008).....	31
<i>Egenera, Inc. v. Cisco Sys., Inc.</i> , 972 F.3d 1367 (Fed. Cir. 2020)	xi, 6
<i>Embrex, Inc. v. Serv. Eng’g Corp.</i> , 216 F.3d 1343 (Fed. Cir. 2000) (Rader, J., concurring)	37
<i>Genentech, Inc. v. Chiron Corp.</i> , 112 F.3d 495 (Fed. Cir. 1997)	41
<i>Glaros v. H.H. Robertson Co.</i> , 797 F.2d 1564 (Fed. Cir. 1986)	50

<i>HVLPO2, LLC v. Oxygen Frog, LLC</i> , 949 F.3d 685 (Fed. Cir. 2020)	54
<i>Jennings v. Jones</i> , 587 F.3d 430 (1st Cir. 2009).....	47
<i>Kaufman v. Microsoft Corp.</i> , 34 F.4th 1360 (Fed. Cir. 2022)	35
<i>Keisling v. SER–Jobs for Progress, Inc.</i> , 19 F.3d 755 (1st Cir. 1994).....	31
<i>Kennedy v. Town of Billerica</i> , 617 F.3d 520 (1st Cir. 2010).....	47
<i>Levesque v. Anchor Motor Freight, Inc.</i> , 832 F.2d 702 (1st Cir. 1987).....	31
<i>Malico, Inc. v. Cooler USA Inc.</i> , 594 F. App’x 621 (Fed. Cir. 2014)	23
<i>Omega Patents, LLC v. CalAmp Corp.</i> , 920 F.3d 1337 (Fed. Cir. 2019)	21, 54
<i>Outside the Box Innovations, LLC v. Travel Caddy, Inc.</i> , 695 F.3d 1285 (Fed. Cir. 2012)	40
<i>People v. Ignacio</i> , 852 F.2d 459 (9th Cir. 1988)	48
<i>Polansky v. CNA Ins. Co.</i> , 852 F.2d 626 (1st Cir. 1988).....	53
<i>Promega Corp. v. Life Techs. Corp.</i> , 875 F.3d 651 (Fed. Cir. 2017)	31
<i>Santiago-Díaz v. Rivera-Rivera</i> , 793 F.3d 195 (1st Cir. 2015).....	30
<i>Smith & Nephew, Inc. v. Ethicon, Inc.</i> , 276 F.3d 1304 (Fed. Cir. 2001)	40

<i>Snuba Int’l, Inc. v. Dolphin World, Inc.</i> , No. 99-1357, 2000 WL 961363 (Fed. Cir. July 11, 2000).....	46
<i>UCB, Inc. v. Watson Lab ’ys. Inc.</i> , 927 F.3d 1272 (Fed. Cir. 2019)	45
<i>Uniloc USA, Inc. v. Microsoft Corp.</i> , 632 F.3d 1292 (Fed. Cir. 2011)	31, 57
<i>United States v. Carpenter</i> , 494 F.3d 13 (1st Cir. 2007).....	52, 53
<i>United States v. Hernandez</i> , 176 F.3d 719 (3d Cir. 1999)	47, 49, 55
<i>Victor v. Nebraska</i> , 511 U.S. 1 (1994).....	50
<i>Wi-LAN, Inc. v. Apple, Inc.</i> , 811 F.3d 455 (Fed. Cir. 2016)	37, 45
<i>Zenith Lab ’ys, Inc. v. Bristol-Myers Squibb Co.</i> , 19 F.3d 1418 (Fed. Cir. 1994)	1, 13, 44

Rules

Fed. R. Civ. P. 37	23
Fed. R. Civ. P. 56	31
Fed. R. Evid. 103	57

REQUEST FOR ORAL ARGUMENT

The issues involved in this Appeal have applicability to the area of patent infringement and the conduct of jury trials. Accordingly, Plaintiff-Appellant Egenera, Inc. believes oral argument is here warranted and requested.

STATEMENT OF RELATED CASES

This matter was previously before this Court in *Egenera Inc. v. Cisco Sys., Inc.*, Case Nos. 2019-2015, 2019-2387. A decision of this Court in that appeal was rendered on August 28, 2020. *Egenera, Inc. v. Cisco Sys., Inc.*, 972 F.3d 1367 (Fed. Cir. 2020).

I. INTRODUCTION

The district court and the jury found noninfringement of a “comprising” claim based on the presence of unclaimed elements found in the accused system. The jury was convinced to wrongfully import these nonexistent elements into the claim by comparing the accused product to the BladeFrame, a commercial product sold by Plaintiff-Appellant Egenera, Inc. (“Egenera”). That type of non-infringement comparison has long since been rejected by this Court. *See, e.g., Zenith Lab’ys, Inc. v. Bristol-Myers Squibb Co.*, 19 F.3d 1418, 1423 (Fed. Cir. 1994) (“As we have repeatedly said, it is error for a court to compare in its infringement analysis the accused product or process with the patentee’s commercial embodiment or other version of the product or process; the only proper comparison is with the claims of the patent.”). When the claims of U.S. Patent No. 7,231,430 (“the ’430 Patent”) are properly construed and applied to Defendant-Appellee Cisco Systems, Inc.’s (“Cisco”) accused Unified Computing System (“UCS”) products, no reasonable jury could conclude that Cisco does not infringe the asserted claims. Vacatur of the district court’s judgment and entry of judgment as a matter of law (“JMOL”) in Egenera’s favor is thus warranted.

To make matters worse, the district court also allowed improper lay opinion testimony, jury instructions, and prejudicial statements to be presented to the jury.

Any one of these errors could support reversal and remand, but the confluence of these errors requires that remedy.

II. JURISDICTIONAL STATEMENT

Egenera appeals from a final judgment of the United States District Court for the District of Massachusetts resolving all claims. The district court had jurisdiction under 28 U.S.C. § 1338(a). Egenera timely appealed and this Court has jurisdiction over this appeal under 28 U.S.C. § 1295(a)(1).

III. STATEMENT OF THE ISSUE PRESENTED FOR APPEAL

1. Whether the district court erred in granting summary judgment that a CPU does not “emulate Ethernet functionality over the internal communication network” where the CPU communicates via a virtual Ethernet connection.

2. Whether a reasonable jury could find that a CPU is not “programmed” to establish a network topology where the CPU is loaded with drivers and a MAC address to establish network communication.

3. Whether a reasonable jury could find that removing a tag from a message does not constitute “modifying” that message.

4. Whether a reasonable jury could find that information used to route and deliver messages to correct locations constitutes an “address.”

5. Whether a new trial is warranted in view of the district court’s uncured incorrect statement of law to the jury panel, the admission of expert

testimony by lay witnesses, and numerous improper and prejudicial statements during closing argument.

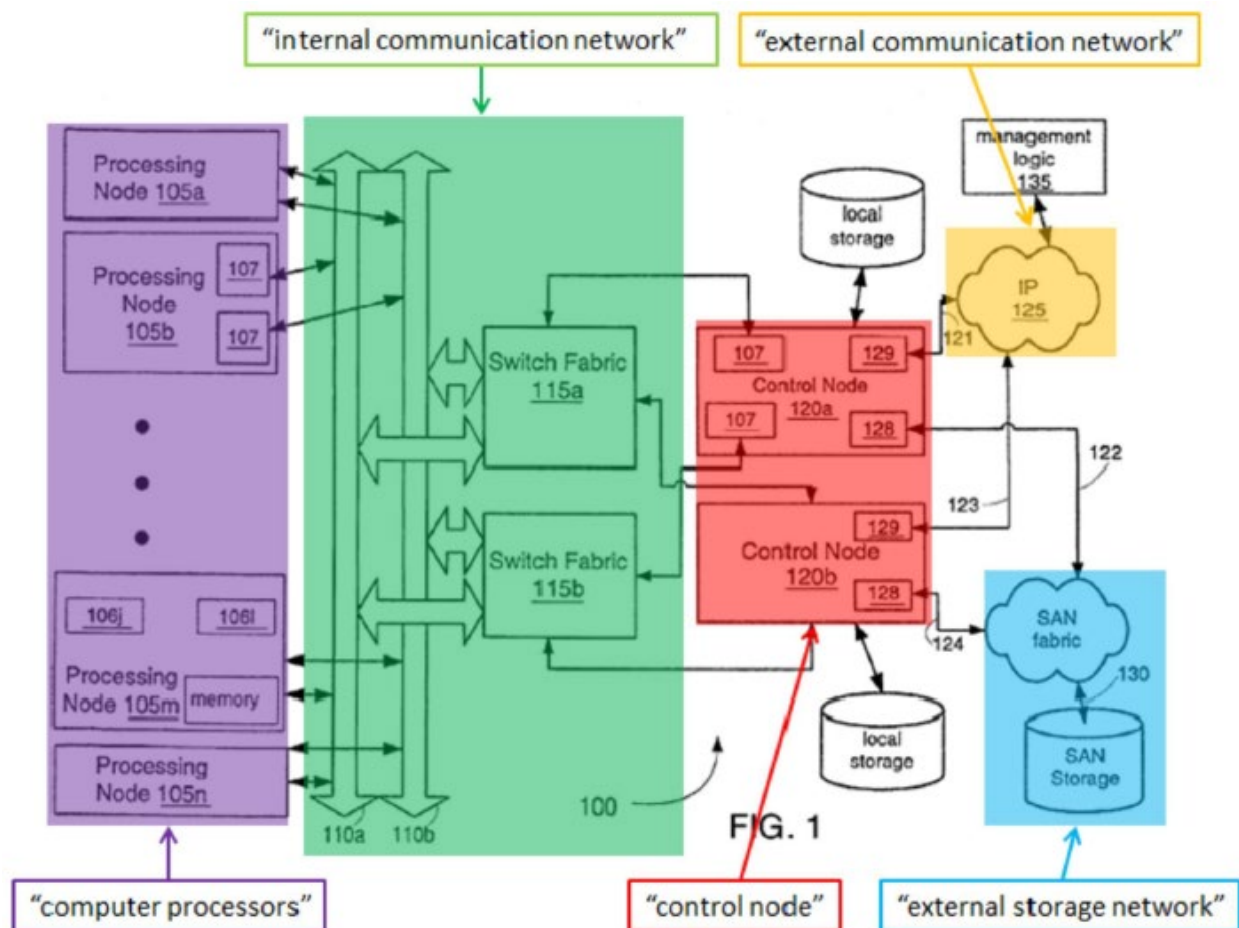
IV. STATEMENT OF THE CASE AND FACTS

A. Egenera and the '430 Patent

Egenera was founded in 2000 by Vern Brownell, a former Chief Technology Officer at Goldman Sachs, an inventor who wanted to simplify data centers. APPX00837. Then-existing enterprise-server systems were inflexible, having to be individually wired and configured, making them difficult to reconfigure. Brownell founded Egenera and assembled a team of experienced engineers to design a server-network architecture that could be wired once and then reconfigured using software and not having to physically rewire the various connections between servers of the network. APPX00839–00845. These server implementations became known “virtual networks” and are now commonplace. But in the early 2000s when Egenera built its original Interframe (later BladeFrame) platform and Processing Area Network Manager software, they were revolutionary. Egenera’s product was widely heralded and adopted by major customers such as Credit Suisse First Boston. APPX00857; APPX00860. These inventions allowed Egenera to grow rapidly, be commercially successful and it remains in business today.

Egenera’s ’430 Patent describes and claims a “Reconfigurable, Virtual Processing System, Cluster, Network and Method.” APPX00086. The claimed

invention enables system administrators to quickly and efficiently deploy virtual processing area networks using software commands. APPX00102 at 1:65-67, APPX00115 at 28:63–29:38. Figure 1, annotated below, shows the main components of the claimed platform and the external networks to which it connects:



APPX00088 (emphasis added). Control nodes 120 (red) are the nerve center for creating and operating the virtual networks. They communicate over an internal network (green) with processing nodes 105 (purple) and with an external

communication network (yellow) and an external storage network (blue). Because the internal and external communications networks may use different protocols, the control node includes logic to modify outbound messages received from processors over the internal network so that those messages will be compatible with and understood by the external network.

Egenera asserted claims 1, 3–5 and 7–8 of the '430 Patent below against Cisco's UCS. The district court granted summary judgment of noninfringement of claims 1 and 5, and Egenera proceeded to trial on claims 3 and 7 only.

B. Claim Construction

The district court entered the following claim constructions relevant to this appeal, with all “logic” terms construed as means-plus-function limitations:

Term	Construction
“computer processor” / “processor” (claims 1, 3, 5 and 7)	“CPU”
“logic to select, under programmatic control, a corresponding set of computer processors from the plurality of computer processors” (claims 1 and 3)	Structure: ““a utility within the management software 135’ and its equivalents.”

“logic to . . . program said corresponding set of computer processors and the internal communication network to establish the specified virtual local area network topology” (claim 3)	Structure: ““control node-side networking logic 310, (reliable) virtual interface 212, and data structure 910’ and their equivalents.”
“logic to . . . program the at least one control node to define a virtual storage space for the virtual processing area network” (claim 3)	Structure: ““storage configuration logic 605, management interface component 610, and storage data structure 815/915’ and equivalents.”

APPX01234–01261. With respect to the claim term “emulate Ethernet functionality over the internal communication network” in claims 1 and 5, the district court declined to limit “internal communication network” to a “non-Ethernet physical network” as proposed by Cisco. APPX01259–01260.

In a previous appeal, this Court affirmed the district court’s holding that “logic” in the claims of the ’430 Patent invokes means-plus-function claiming. *Egenera*, 972 F.3d at 1375.

C. Cisco’s UCS and Noninfringement Arguments

It was undisputed throughout the proceedings that Cisco’s UCS products have the basic architecture described and claimed in the ’430 Patent. Rather,

Cisco’s noninfringement arguments focused primarily on the use of network interface cards (“NICs”) in its accused system. A NIC is a hardware component distinct from but communicating with the CPU and that enables network communication.

As an initial matter, the ’430 Patent explicitly contemplates the use of NICs in the claimed system, explaining:

Under certain embodiments, about 24 processing nodes 105*a-n*, two control nodes 120, and two switch fabrics 115*a, b* are contained in a single chassis and interconnected with a fixed, pre-wired mesh of point-to-point (PtP) links. Each processing node 105 is a board that includes one or more (e.g., 4) processors 106*j-l*, one or more network interface cards (NICs) 107, and local memory (e.g., greater than 4 Gbytes) that, among other things, includes some BIOS firmware for booting and initialization.

APPX00103 at 3:9-17. The ’430 Patent thus does not preclude the use of NICs, but the asserted claims do not *require* them. Nonetheless, Cisco relied extensively at trial on its NICs in seeking to escape infringement of the ’430 Patent.

1. Cisco’s Summary Judgment Noninfringement Position

Claims 1 and 5 require that the claimed plurality of computer processors “emulate Ethernet functionality over the internal communication network.”

Specifically, Cisco admitted that the UCS did emulate Ethernet functionality but argued that the emulation functionality was exclusively performed by NICs rather

than the CPUs. Egenera countered the CPUs are heavily involved in the emulated Ethernet network and in fact participate in the claimed emulation. In particular, Egenera showed that Cisco's UCS associates a service profile with a Server and its CPUs, creating a virtual connection between CPUs and Fabric Interconnects. APPX05223. Egenera then explained that the UCS Server CPU's Service Profile includes a virtual MAC address, enabling the CPU to communicate over the virtual Ethernet network. In this way, the CPU includes logic to emulate Ethernet functionality, because the CPU can perform Ethernet functionality in a virtual Ethernet network.

The district court dismissed this evidence, granting summary judgment of noninfringement as to claims 1 and 5, leaving only claims 3 and 7 to be presented at trial.

2. Cisco's Noninfringement Positions at Trial

For claims 3 and 7, too, Cisco emphasized its NICs to show noninfringement. Cisco's defense was based on what Cisco's counsel called "three bedrock facts" that Cisco would set out to prove during the trial. APPX12147, 66:22-25. As made clear in Cisco's counsel's opening statement, Cisco's defense was based on the assertion that the UCS programs the NIC rather than the CPU:

Bedrock fact number one: The Cisco UCS did not cause the BladeFrame to fail. The BladeFrame failed in the market on its own because its design didn't work. . . .

Bedrock fact number two: This patent claim requires that you set up a network by programming the CPUs, and that element was added to the claim in a meeting at the patent office, and they wouldn't have a patent if that wasn't in the claim. . . .

Bedrock fact number three: The Cisco UCS does not set up the network by programming the processors. We program the NICs, we designed the NIC ourself. The NIC is not a CPU. We do not infringe this claim. . . .

APPX12148, 67:1-14. Cisco raised a total of three specific noninfringement arguments at trial. Its CPU's alleged programing of the NICs to establish a network topology was the centerpiece of its defense.

a. Cisco's First and Centerpiece Noninfringement Argument: Programming the CPUs to Establish a Network Topology

Claims 3 and 7 each require programming a set of CPUs to establish a virtual local area network topology. Again, it was undisputed that the UCS performs *some* programming to set up a virtual LAN topology. Cisco thus focused on the fact that a *portion* of the programming occurs in the NICs rather than the CPUs.

Egenera presented *unrebutted* evidence at trial that, while some of the topology programming occurs in the NICs, topological programming *also* occurs in the CPUs as required by the claims. For example, Egenera showed that the UCS server CPUs must be loaded with the appropriate drivers and MAC addresses

before the CPUs can communicate with other CPUs in the network. These functions were critical to creating a network topology and, critically, that testimony stands unrebutted. Loading this software and these address data constitutes “programming” the CPUs. Indeed, the network topology does not exist until the CPUs can communicate with each other, logically and inescapably meaning that the CPUs of the accused products are programmed by Cisco to establish the claimed network topology.

Specifically, Cisco’s Vice President, Bhaskar Jayakrishnan, admitted that a CPU in the UCS cannot communicate with a CPU in another server until after each CPU is programmed with the appropriate drivers and MAC address:

Q: So the message cannot be sent from the first CPU if that first CPU is not running the OS; correct?

A: Yes, that is correct. For that CPU to do anything it needs an operating system.

. . . .

Q: And if that CPU doesn’t have the right drivers for that adapter card it also can’t send the message; correct?

A: It would not. The operating system would not know how to use that network card. That is correct.

APPX12896, 123:12-21; *see also* APPX12898–12901, 125:9–128:2. Mr.

Jayakrishnan also admitted that until the processors are programmed with the drivers and MAC address from the virtual NICs, one CPU simply cannot send a

message to another CPU in the system. APPX12898–12901, 125:9–128:2. The relevant operation of UCS is thus undisputed—there is no interconnectivity among the claimed processors, and thus no network topology, until the accused processors are programmed with the drivers and the MAC addresses from the virtual NICs, and that programing is admittedly done by Cisco’s accused product.

Egenera’s expert, Dr. Mark Jones, confirmed this understanding, testifying that the processors are programmed to establish the network topology “when the operating system . . . gathers information about the devices, the peripherals that it has, and stores that in the operating system. At that point, it’s programmed with that information.” APPX12696, 77:15-22; *see also* APPX12696–12705, 77:23–86:15; APPX12715–12717, 96:20–98:11. Dr. Jones then explained that this programing includes loading onto the CPUs the operating system and device drivers necessary for the CPUs to communicate.

For example, when asked whether, before being so programmed, a given CPU could communicate with other CPUs in the network, Dr. Jones (like Mr. Jayakrishnan) explained “[n]o, it cannot. It doesn’t have its operating system loaded, for example. It’s not booted up. But it also doesn’t know how — it doesn’t have the information it needs to communicate with another processor.” APPX13166–13167, 68:25–69:5. Dr. Jones further explained that, even after being loaded with an operating system, the CPU still cannot communicate with any other

CPU because “[t]he service profile hasn’t been fully associated and the CPU, including the OS, does not have the information it needs to communicate with, for example, communicate through the VNIC.” APPX13167–13168, 69:25–70:5. As Dr. Jones explained, the CPU may communicate with another CPU only after the appropriate drivers are loaded: “that virtual network interface card, it will load what’s known as device drivers. . . . Device drivers are how an operating system talks to a physical device, so there is a device driver in software, in UCS for talking to the VNICs.” APPX12696–12697, 77:24–78:21.

Further, Dr. Kevin Jeffay—Cisco’s technical expert—did not address at all any of Dr. Jones’ or Mr. Jayakrishnan’s testimony that the CPUs must be programmed with drivers before they can communicate. *See* APPX12973–12982, 33:22–42:23. Instead, for this limitation, Dr. Jeffay opined only that programming processors and programming NICs are “fundamentally different” approaches, without addressing that both the NIC and the processor are programmed in UCS. *See* APPX12974, 34:1-4.

Rather than refute this evidence (which it could not do), Cisco sought to mislead the jury on this issue by pointing to differences between the accused UCS product and Egenera’s BladeFrame product. For example, Dr. Jeffay testified to the ability to “swap out” a processor as a basis for distinguishing between Egenera’s product and Cisco’s product. *See* APPX12956–12959, 16:14–19:5.

Although Dr. Jeffay did not testify that this difference was evidence of noninfringement—which it is not—Cisco’s counsel argued during closing argument that allowing a user to “swap out for a failed CPU” is a “fundamental” aspect of the invention and a basis for Cisco’s noninfringement. *See, e.g.*, APPX14963–14964, 10-25:18–10-26:10 (“Cisco does not infringe the ’430 Patent. . . . It says right in the patent description if the CPU fails, you can swap out another. You can’t do that in the Cisco UCS, and nobody disputes that. . . . The UCS doesn’t work like the Egenera product. It just doesn’t and it can’t.”). But the ability to “swap out” a failed CPU is not required by any claim of the ’430 Patent. Most importantly, though, claiming noninfringement based on comparing competing product features, as opposed to comparing the accused product features to the properly construed claim terms, constitutes black letter legal error. *Zenith*, 19 F.3d at 1423.

**b. Cisco’s Second Noninfringement Argument:
Modifying and Transmitting Received Messages**

Cisco next argued that the UCS does not meet the requirement to “modify said received messages to transmit said modified messages to the external communication network and to the external storage network” in claims 3 and 7. But Egenera provided substantial un rebutted evidence that Cisco’s UCS Fabric Interconnect, a component of the UCS that corresponds to the claimed “control

node,” receives messages from the computer processors and, before sending them out to the external communication network, modifies those messages by removing a tag called the “VN-Tag.” APPX12661–12662, 42:7–43:1, APPX12665–12667, 46:18–48:1. Cisco did not dispute Egenera’s evidence of the operation of the system. *See, e.g.*, APPX12937, 164:18-23 (Dr. Jeffay testifying, “I think some of the factual nuts and bolts of how it works, I think what he said is accurate. It’s just I disagree with the conclusions that he draws as a result of that.”); *see also* APPX13008–13009, 68:23–69:7 (Dr. Jeffay agreeing that Dr. Jones packet captures showing removal of a VN-Tag “were correct”). Rather than dispute the operation of UCS, as it could not, Cisco instead tried to distract the jury with two legally insufficient bases for ignoring the plain language of the claims.

First, Cisco argued that removal of the VN-Tag cannot be the claimed modification because the patent requires translating from Giganet to Ethernet and, because UCS does not use Giganet on its internal network, it therefore does not need to modify messages to use Ethernet on the external network. APPX13002, 62:16-23, APPX13004–13005, 64:9–65:10 (“[I]f a processor wants to send data to some client on the internet, it will first generate a Giganet message . . . [and] the control node again has to do a form of translation or modification of the message to get it into the form of Ethernet.”); APPX13002–13003, 62:24–63:6 (“Q And what is the claim language that this element relates to? A It relates to . . . logic to receive

messages from computer processors and then modify the received messages to transmit the modified messages to the external communication network.”). But claims 3 and 7 do not require translation from Giganet to Ethernet, and Cisco’s expert admitted this very point. APPX13041, 101:14-16. Accordingly, this could not be a basis for finding non-infringement.

Second, Cisco argued that removal of the VN-Tag is not the claimed modification because there is an additional step requiring message modification to occur before they reach the Fabric Interconnect. APPX13009, 69:8-23. Dr. Jeffay did not dispute that messages are modified by the UCS Fabric Interconnect. *Id.* at 69:13 (“[T]he fabric interconnect takes off the tag.”). The fact that additional modifications to the message may take place at other points in the overall message transmission process has no bearing on whether the UCS Fabric Interconnect receives and modifies messages. It does and that is all the claims require. Dr. Jones provided un rebutted expert testimony that “the removal of the VN tag is done at the control node, and that’s what the claim requires, and the messages are coming from the processors.” APPX13176–13177, 78:24–79:8. Again, then, this argument too could not be the basis for the jury finding non-infringement.

**c. Cisco's Third Noninfringement Argument:
Extracting an Address and Identify a Corresponding
Address**

Cisco finally argued that the UCS does extract an address from messages and identify a corresponding address. APPX12996–12997, 56:6–57:10. But Egenera provided substantial un rebutted evidence that the UCS Fabric Interconnect, which again, is the claimed “control node,” extracts an address from messages destined for the external storage network, identifies a corresponding address, and then sends that message out to the external storage network as required by the claims. *See, e.g.*, APPX12711–12713, 92:7–94:11.

In particular, Dr. Jones explained the Fabric Interconnect extracts a “VLAN ID” and identifies the corresponding “VSAN ID” number for transmission of the message to the external storage network. APPX12711–12712, 92:23–93:14. Further, Dr. Jones explained that the Fabric Interconnect also extracts an address during the conversion of FLOGI to FDISC for messages sent to the external storage network:

[A]s we saw earlier in the structure for the FI [Fabric Interconnect], in examples up here, it extracts information including header information such as addresses. This is the storage message going out to the storage network. It will identify a corresponding address for which this message should be sent and it's going to be sent out over a particular port, a port associated with that external storage network and will provide messages to that storage network over that particular external port.

APPX12712–12713, 93:18–94:1.

As with message modification, Cisco did not dispute the operation of the UCS, but rather attempted to distinguish UCS from Egenera’s commercial BladeFrame product, arguing that the UCS cannot meet this limitation because it does not have “a centralized I/O gateway” that was similar to Egenera’s BladeFrame product. APPX12996–12997, 56:6–57:17. But the claims do not require a centralized I/O gateway. APPX00116, 29:41–30:28; APPX00117, 31:65–32:53. The claims require extracting an address from a message and identifying a defined corresponding address in the external storage space. That is all and the accused UCS admittedly performs that function.

Similarly, Dr. Jeffay’s further non-infringement arguments for this limitation failed to address the actual requirements of the claim and instead focused on the fact that UCS does not “translate” storage addresses: “The claims are going to require that you translate the address.” APPX12986, 46:11-19. “[In UCS, t]here’s no address translation required, so there’s no identifying of the corresponding storage address.” APPX12993, 53:9-10. “So because it’s using this protocol, fibre channel over Ethernet, [the UCS] does not need to translate addresses, and therefore, it doesn’t perform the step of identifying a corresponding address.” APPX12995, 55:2-5. “[Y]ou had to have this centralized I/O gateway in the control node to translate addresses, and the address translation does not happen,

and it's not required in UCS." APPX13000–13001, 60:22–61:1; *see also* APPX12983–13001, 43:5–61:21. The claims do not require any such address "translation."

But, even if the claims did require address translation, which they do not, Dr. Jones provided substantial un rebutted expert testimony that translation occurred. APPX12711–12712, 92:23–93:14; *see also* APPX13175, 77:13–23 ("What I'm describing is the translation, and I indicated how it operates, including how an address is extracted from FLOGI, and that is used to find the corresponding port which is used to forward the information.")). Indeed, Dr. Jeffay failed entirely to address this testimony from Dr. Jones regarding the extraction of an address during the FLOGI to FDISC conversion process. Instead, Dr. Jeffay "rebutted" an argument Dr. Jones never made, arguing that FLOGI and FDISC are not addresses. *See* APPX12985, 45:12–15 ("I've highlighted the FLOGI versus FDISC because this message can't satisfy this limitation because neither FLOGI or FDISC are actually addresses.")). But Egenera did not argue that FLOGI and FDISC themselves are addresses, but rather that VLAN ID and VSAN ID are addresses. Cisco's trial evidence did not and could not refute this point.

3. The Conduct of Trial

The case proceeded to trial in August 2022 on the issues of infringement, validity, willfulness and damages. The jury found the patent valid and not

infringed, meaning neither willfulness nor damages were reached. Egenera does not appeal the jury's finding of validity.

At trial, Cisco's primary noninfringement theme was to distinguish the accused UCS from the BladeFrame rather from the claims, asserting that it did not *infringe* the '430 Patent because it did not *copy* the BladeFrame. *See, e.g.*, APPX14984, 10-46:18-19 ("So we didn't copy the 2004. We didn't copy the 2008. We didn't copy the BladeFrame."). To be sure, Egenera *did* accuse Cisco of copying the '430 Patent and the BladeFrame product as part of its case for *willful* infringement. But the question of whether Cisco *infringed* the '430 Patent had nothing to do with copying, and Cisco's counsel's improper conflation of those issues led to jury confusion.

The conflation problem began on day 1, with the district court providing a "brief description of the case" to orient the prospective jury to the parties and the nature of the case, including the concept of patent infringement. APPX13443-13445, 11:19-13:6. The district court's summary departed in large part from agreed-upon materials that the parties submitted in their Joint Pretrial Memorandum. APPX11428-11483; APPX11494-11501; APPX11503-11505. In its summary, the district instructed the jury that "to 'infringe' in the patent context means to *copy* essentially without permission." APPX13444, 12:18-19 (emphasis added). This instruction was directly contrary to law, as "copying . . . is of no

import on the question of whether the claims of an issued patent are infringed.”

Allen Eng’g Corp. v. Bartell Indus., Inc., 299 F.3d 1336, 1351 (Fed. Cir. 2002).

Egenera promptly requested a curative instruction that would have informed the jury the district court had “inadvertently explained the concept of direct patent infringement” to mean “to copy essentially without permission.” APPX11814-11819. The proposed instruction would then have informed the jury that they “do not need to determine whether Cisco copied the patented technology to determine whether Cisco directly infringes the ’430 patent.” APPX11815–11816. The district court did not include Egenera’s proposed curative instruction in the final jury instructions. Egenera preserved its objection to that omission before and after the jury charge. APPX11983–11984; APPX15075, 10-137:14-18.

The confusion and conflation of patent infringement with copying infused the trial. Cisco also elicited expansive non-infringement opinions from two Cisco lay witnesses, Michael Dvorkin and Mr. Jayakrsihnan. Egenera sought to prevent this testimony with a motion *in limine*, APPX10917, ¶ 3, which the district court denied under *Omega Patents, LLC v. CalAmp Corp.*, 920 F.3d 1337, 1352–53 (Fed. Cir. 2019), ruling that “the opinion testimony [of Cisco’s two lay witnesses] is relevant to intent and knowledge for purposes of defending the willful infringement and indirect infringement claims.” APPX00165 at ECF 411 ¶ 3. But Mr. Dvorkin’s and Mr. Jayakrishnan’s testimony far exceeded what is allowed

under *Omega*. In effect, they became surprise expert witnesses for Cisco, despite not being qualified as experts in this case or submitting expert reports.

For example, Mr. Dvorkin directly addressed and disputed expert testimony given by Egenera's expert, Dr. Jones. Regarding the programming of the CPUs, Mr. Dvorkin improperly opined that Dr. Jones was wrong to conclude that "UCS Manager programmed server CPUs to establish the network topology," because Cisco "programmed the interface card." APPX14543, 131:5-10. Mr. Dvorkin, without explanation, also disagreed with Dr. Jones' conclusion that CPUs "discover[ing] the information about the network adapters as peripheral devices in the servers . . . means UCS programs the CPUs," *id.* at 131:14-20, further opining that Cisco's purported programming of the CPUs was "very different" from "Egenera's patented design" and "not a copy" of Egenera's design. APPX12795, 22:2-7. Mr. Dvorkin also opined that Cisco's litigation-generated "bedrock fact number three," directed at non-infringement, was true in its entirety. APPX14544–14545, 132:21–133:23. Mr. Dvorkin also addressed the doctrine of equivalents—over Egenera's specific objection, APPX12815–12816, 42:24–43:2-7—opining that "programming a UCS NIC to establish the network topology" was "very different" from "programming the UCS server CPU to establish the network topology," APPX12816, 43:8-14, largely because "Egenera's approach is done

within a host within the operating system, the specialized software that emulates I/O functions.” APPX12816–12817, 43:21–44:5.

Mr. Jayakrishnan’s opinion testimony mirrored in many respects that of Mr. Dvorkin. Mr. Jayakrishnan disputed portions of Dr. Jones’ expert testimony, disagreeing with Dr. Jones’ opinion that the NICs were not independent devices, but rather operated under the CPUs, and employing an analogy to the standby mode in televisions. APPX12865–12866, 92:3–93:12, APPX12868–12869, 95:15–96:6. Mr. Jayakrishnan also testified that the CPU could ask the network card for its MAC address did not mean “that the CPUs [are] programmed with the network topology,” as required by the patent claims. APPX12877, 104:7-18. And, just as Mr. Dvorkin had done, Mr. Jayakrishnan opined that each statement in Cisco’s bedrock fact number three regarding noninfringement was true. APPX12873–12874, 100:9–101:9. Mr. Jayakrishnan also opined, as did Mr. Dvorkin, that programming the NICs in the UCS is not substantially the same as programming the CPUs, disputing Dr. Jones’ opinion to the contrary. APPX12880–12881, 107:22–108:5. According to Mr. Jayakrishnan, the two are substantially different because it is possible to “program[] the NIC even when the CPUs don’t even have any power.” *Id.* Mr. Jayakrishnan further elaborated by stating that “the network topology is set in the NIC” such that “when the CPU boots up it will learn a lot

about the network topology by asking the NIC, but the programming of the NIC is already done.” *Id.*

As a particularly egregious example of lay expert testimony, Mr. Jayakrishnan described a litigation-inspired test protocol purportedly intended to confirm that the UCS network topology is established on the NIC rather than the CPU. APPX12869–12872, 96:7–99:1. According to Mr. Jayakrishnan, he and his team performed this test and verified Mr. Jayakrishnan’s understanding. APPX12871, 99:10-15. Notably, the results of this test were not provided to Egenera at any point during the litigation, including in any expert report, and thus these materials should have been excluded at trial. *See, e.g., Malico, Inc. v. Cooler USA Inc.*, 594 F. App’x 621, 625 (Fed. Cir. 2014) (affirming exclusion of evidence that had not be produced in discovery); Fed. R. Civ. P. 37(c)(1) (“If a party fails to provide information or identify a witness as required by Rule 26(a) or (e), the party is not allowed to use that information or witness to supply evidence on a motion, at a hearing, or at a trial, unless the failure was substantially justified or is harmless.”).

The confusion between what patent claims require and a commercial products was exacerbated by the district court’s refusal to include Egenera’s

proposed jury instruction that a patented product may still infringe another patent.¹ Indeed, a second persistent theme of Cisco’s trial defense was that it could not infringe the ’430 Patent because Cisco had obtained its own patents on technology incorporated into the UCS.² To mitigate against potential juror confusion and prejudice to Egenera from common misconceptions surrounding patents, Egenera proposed a simple jury instruction: “A product that is covered by a subsequent or later patent may still infringe an earlier patent.” APPX11456. The district court declined to include the requested instruction in the final jury instructions, and Egenera objected. APPX11977–11978; APPX13529, 15:17–16:17; APPX15075, 10-137:14-18, *see also generally* APPX15030–15077, 10-92:7–10-139:5.

But Cisco’s improper trial conduct did not end there. During closing argument, Cisco’s counsel made arguments that were contrary to the district court’s rulings and established law. Prior to trial, the district court granted four motions *in limine* relevant here. By agreement of the parties, the district court prohibited “[t]estimony, argument, or reference to the absence of any witnesses

¹ Egenera preserved its objection to the omission of this instruction. APPX11977–11978; APPX15075, 10-137:14-18.

² APPX12796–12797, 23:1–24:10; APPX13607; APPX12857, 84:20-24; APPX12990–13002, 50:2–62:12; APPX13006–13010, 66:15–70:23; APPX13741–13743; APPX13609–13627; APPX13629–13646; APPX13648–13672; APPX13674–13707; APPX13709–13721; APPX13723–13733; APPX14962, 10-24:7-11, APPX14975–14976, 10-37:12–10-38:11.

who do not appear at trial,” APPX00168–00169, ECF 431, ¶ 12—*i.e.*, the district court prohibited “empty chair” arguments. The district court, also by agreement, prohibited Cisco both from “referring to Egenera as a non-practicing entity or a patent troll,” and from “making any arguments that . . . non-practicing entities bring baseless claims.” *Id.*, at ECF 431, ¶¶ 2–3. Fourth, the district court granted Egenera’s opposed motion *in limine* to “preclude the parties from referencing either Egenera’s or Cisco’s ability to finance the present litigation, including through litigation funding or investment by private equity funders.” APPX11558; *see also* APPX00166, ECF 412. Egenera sought with this motion to prevent suggestion at trial that “Egenera’s patent infringement claim is predatory or lacks merit, or that Egenera should not otherwise be entitled to meaningful damages or other relief.” APPX11562. The district court granted this later motion *in limine* in its entirety and with no conditions. APPX00166, ECF 412.

Cisco violated each of these orders during its closing arguments. Cisco’s counsel, for example, violated the “empty chair” prohibition by arguing that Egenera did not call Mike Thompson, Egenera’s former CEO, live at trial because Egenera did not want the jury to hear the truth:

[Mr. Thompson] was the CEO in 2008. He’s one who knows, and they didn’t call him. . . . Why did they bring the other two guys to tell you HP wasn’t a competitor and they were losing sales to UCS and they didn’t bring the

guy that was there at the controls when they canceled the BladeFrame? Why? Because he told the truth.

APPX14952–14953, 10-14:17–10-15:1. Cisco’s counsel also leveraged the empty chair to leave the jury with the false impression that Egenera no longer has employees. APPX14986, 10-48:23-25 (attacking Egenera for bringing “no witness here to talk about Egenera today” and for not proving at trial whether “anyone still work[s] there”), APPX14987, 10-49:1-6 (“No one testified about Egenera today . . . We don’t [know] if there’s anyone left at Egenera.”).

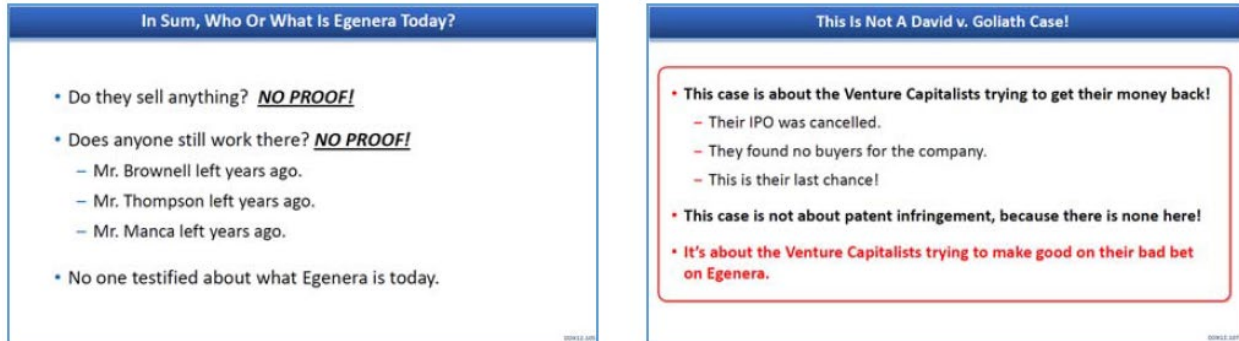
Cisco’s counsel also violated the district court’s litigation finance order—an order sought for the specific purpose of preventing Cisco from improperly suggesting that “Egenera’s patent infringement claim is predatory or lacks merit, or that Egenera should not otherwise be entitled to meaningful damages or other relief.” APPX11562. Cisco’s counsel demonized Egenera’s investors as “venture capitalists swarming around, who put money in Egenera, and are sad that they lost it,” insinuating that the lawsuit was an investor-funded “last attempt to make some money on that bet.” APPX14987, 10-49:7-14; *see also id.* at 10-49:19-24 (“This case is not about patent infringement” but rather about “a bunch of venture capitalists trying to make good on their bad bet.”); APPX14988, 10-50:1-2 (“They’re stuck with a bad investment, and they’re looking for an exit.”). Cisco’s counsel further inappropriately called out by name Egenera’s corporate

representative, Jim Phillips, arguing without evidence (and incorrectly) that “Mr. Phillips gets the money. He’s a venture capitalist who bet on Egenera, and he’s upset because he lost his money.” APPX14987, 10-49:7-10. There was no evidence at all to support these inflammatory comments and insinuations.

In addition, Cisco violated the district court’s orders prohibiting Cisco “from referring to Egenera as a non-practicing entity or a patent troll.” APPX00168, ECF 431, ¶ 2. Indeed, Cisco questioned whether Egenera was still an operating company and further argued that Egenera no longer used the ’430 Patent, effectively characterizing Egenera as a non-practicing entity. APPX14956, 10-18:8-12 (“[W]hen they stop[ped] selling the BladeFrame they stopped using their patent.”); *id.* at 10-18:20-25 (“And at that time in 2008, they stopped using their own patent. Think about that.”). Taken together with Cisco’s counsel’s gratuitous arguments that this was just a case about making good on a “bad bet,” APPX14987, 10-49:7-24, APPX14988, 10-50:1-4, Cisco effectively violated the district court’s order prohibiting “any arguments . . . that non-practicing entities bring baseless claims.” APPX00168, ECF 431, ¶ 3.

The closing arguments outlined above are improper because they disregarded rulings of the district court, distracted jurors from the questions properly before them (infringement and damages), and misstated facts about Egenera. For added effect, Cisco’s counsel coupled his improper oral arguments

with visual demonstratives for which Egenera was not given advance notice or opportunity to object:



APPX13746–13747.

D. Post-Trial Proceedings

Egenera moved post-trial for JMOL that the UCS infringes claims 3 and 7 of the '430 Patent and for a new trial.

The district court denied both motions in a brief eight-page order.

APPX00077–00084. First, the district court dismissed Egenera's argument that "the only basis on which the jury could have found non-infringement is by importing additional limitations into the claims." APPX00078. In a single paragraph of reasoning, the district court simply paraphrased Cisco's noninfringement arguments with little discussion. APPX00078–00079. The district court dispatched Egenera's motion for a new trial with comparably perfunctory analysis. APPX00079–00083.

V. SUMMARY OF ARGUMENT

The Court should vacate the district court's summary judgment of noninfringement of claims 1 and 5 because a reasonable jury could find that a CPU that communicates via a virtual Ethernet network "emulate[s] Ethernet functionality over [an] internal communication network." The district court's factual finding to the contrary at the summary judgment stage was improper.

The Court should reverse the district court's denial of JMOL as no reasonable jury could find noninfringement. The UCS CPUs are programmed with drivers and MAC addresses, and no network exists until the CPUs are so programmed, meaning the limitation "program said corresponding set of computer processors and the internal communication network to establish the specified virtual local area network topology" was met.

Regarding the claim limitation "modify said received messages to transmit said modified messages to the external communication network and to the external storage network," the UCS products receive messages from the CPUs and remove tags from those messages before transmitting the messages to an external communication network. This tag removal is a modification of the messages and again the limitation was met by the accused product.

Finally, it was undisputed at trial that the UCS products extract a "VLAN ID" and identify the corresponding "VSAN ID." These values constitute address

information and meet the “extract an address from a received storage message, to identify the defined corresponding address in the external storage address space” limitation.

Finally, barring entry of JMOL in Egenera’s favor, the Court should order a new trial because (1) the weight of the evidence favors Egenera; (2) the district court improperly referenced “copying” in relation to patent infringement during jury empanelment; (3) the district court allowed copious lay opinion testimony regarding infringement, including an undisclosed “test” performed by one of Cisco’s fact witnesses; and (4) Cisco raised numerous irrelevant, misleading, and inflammatory arguments during its closing.

VI. ARGUMENT

A. Standard of Review

In reviewing the grant of a motion for summary judgment, this Court applies the law of the regional circuit in which the district court sits, here, the First Circuit. *AbbVie Deutschland GmbH & Co., KG v. Janssen Biotech, Inc.*, 759 F.3d 1285, 1295 (Fed. Cir. 2014). The First Circuit reviews a grant of summary judgment *de novo*. *Santiago-Díaz v. Rivera-Rivera*, 793 F.3d 195, 199 (1st Cir. 2015). Summary judgment is appropriate only “if the movant shows that there is no genuine dispute as to any material fact and the movant is entitled to judgment as a matter of law.” Fed. R. Civ. P. 56(a).

This Court “review[s] a district court’s rulings on post-trial motions for JMOL and a new trial under regional circuit law.” *Promega Corp. v. Life Techs. Corp.*, 875 F.3d 651, 659 (Fed. Cir. 2017). “The First Circuit reviews a district court’s denial of JMOL after a jury verdict de novo, asking whether ‘the evidence points so strongly and overwhelmingly in favor of the moving party that no reasonable jury could have returned a verdict adverse to that party.’” *Uniloc USA, Inc. v. Microsoft Corp.*, 632 F.3d 1292, 1301 (Fed. Cir. 2011) (quoting *Keisling v. SER–Jobs for Progress, Inc.*, 19 F.3d 755, 759–60 (1st Cir. 1994)).

The First Circuit reviews the denial of a motion for new trial for an abuse of discretion. *Davignon v. Hodgson*, 524 F.3d 91, 100 (1st Cir. 2008). The Court may reverse a district court’s denial of a motion for new trial “if ‘the verdict is so seriously mistaken, so clearly against the law or the evidence, as to constitute a miscarriage of justice.’” *Astro-Med, Inc. v. Nihon Kohden Am., Inc.*, 591 F.3d 1, 13 (1st Cir. 2009) (quoting *Levesque v. Anchor Motor Freight, Inc.*, 832 F.2d 702, 703 (1st Cir. 1987)).

B. The District Court Erred as a Matter of Law in Entering Summary Judgment of Noninfringement of Claims 1 and 5.

Cisco sought summary judgment of noninfringement of claims 1 and 5 on the grounds that the limitation “the plurality of computer processors . . . include network emulation logic to emulate Ethernet functionality over the internal

communication network” in claim 1 and the limitation “the plurality of computer processors . . . emulate Ethernet functionality over the internal communication network” in claim 5 are not met by the UCS. The district court granted Cisco’s motion on a narrow and erroneous basis.

As the district court noted, “Cisco does not dispute that UCS emulates Ethernet functionality (at least for purposes of this motion) but contends that because Ethernet emulation functionality resides with virtual network interface cards (NIC) and interfaces . . . the limitations are not met.” APPX00048. The district court further agreed with Egenera that because the Ethernet functionality in claims 1 and 5 is attributed to “the plurality of computer processors *and at least one control node,*” “the emulation functionality is not required to reside uniquely on the CPUs.” *Id.* (emphasis in original). But the district court nonetheless rejected Egenera’s argument that UCS Server CPUs satisfy the claim limitation because they “communicate on and use virtual interfaces between themselves and UCS Fabric Interconnects over the UCS internal communication network.” APPX00048—00049, quoting APPX05222.

In a single paragraph of reasoning, the district court made an improper factual finding that “knowledge and use of a communications network is not emulation of the functionality of that network.” APPX00049. The district court’s only support for this conclusion is an inapposite analogy that “a person dialing and

making a telephone call to another's phone number merely uses a telephone network and does not emulate any functionality of that network." *Id.* The district court's telephone analogy does not support its conclusion and demonstrates why its holding is legally wrong. The CPUs are not equivalent to "a person dialing and making a telephone call" as the district court held. The CPUs do not merely *use* the virtual Ethernet; they help to create the desired network.

In any case, the district court ignored Egenera's evidence that the CPUs include Ethernet emulation functionality: "UCS Server CPUs communicate on and use virtual interfaces between themselves and UCS Fabric Interconnects over the UCS internal communication network. The UCS Server CPU communications on (and uses of) these virtual interfaces satisfy the CPU-related Ethernet emulation functionality required in the claims." APPX05222 (internal citations omitted). Egenera then explained that the association of a service profile with a UCS Server *and its CPUs* creates a virtual connection between CPUs and Fabric Interconnects, and that this virtual connection includes virtual interfaces at the Fabric Interconnect and at the UCS Server. APPX05223. Further, the virtual interface functionality includes assignment of virtual MAC addresses to UCS Servers, which are provided to the CPUs within each Server. *Id.* The virtual MAC address being provided to the CPUs is what enables the emulated Ethernet functionality in the

UCS, and thus the CPUs themselves participate in Ethernet emulation.

APPX05223–05224.

The district court ignored this explanation and incorrectly determined that the CPU’s role is limited to “knowledge and use of the virtual MAC address.” APPX00049. In fact, the Ethernet emulation cannot occur without the assignment of the MAC addresses to the CPUs. A reasonable jury could therefore conclude that the challenged limitations of claims 1 and 5 are met, and the district court’s factual finding to the contrary was improper.

The district court also ignored the fact that the UCS operates consistently with the embodiments of the ’430 Patent itself. Like the embodiments described in the ’430 Patent specification, Cisco’s UCS emulates Ethernet functionality using virtual components and virtual interfaces (such as virtual NICs, or “vNICs”) as well as virtual MAC addresses. In particular, the ’430 Patent specification states that “[e]ach PAN, through software commands, is configured to have a corresponding subset of processors 106 that may communicate via a virtual local area network that is emulated over the PtP mesh. . . . No physical deployment or cabling is needed to establish a PAN.” APPX00103, 3:55-60.

The ’430 Patent specification further states that “[u]nder certain preferred embodiments, software logic executing on the processor nodes and/or the control nodes emulates switched Ethernet semantics.” *Id.* at 3:60-63. In other words, the

specification clearly describes CPUs *communicating* over an emulated Ethernet network and characterizes that communication as “emulat[ing] switched Ethernet semantics.” The specification also explicitly contemplates Ethernet emulation being executed both by CPUs and control nodes. The district court’s ruling effectively excluding this functionality from the scope of claims 1 and 5 was thus legal error, as “[a] claim construction that excludes a preferred embodiment is rarely, if ever correct.” *Kaufman v. Microsoft Corp.*, 34 F.4th 1360, 1372 (Fed. Cir. 2022) (internal quotation omitted).

At a minimum, the question of whether the UCS infringes claims 1 and 5 should have been presented to the jury. The district court committed legal error in improperly construing the scope of claims 1 and 5 to exclude a preferred embodiment, and improperly resolved a factual issue in deciding that the UCS does not meet the emulated Ethernet limitations. Vacatur and remand on this basis is thus appropriate.

C. The District Court Erred in Denying Egenera’s JMOL Motion.

The district court should have entered JMOL that claims 3 and 7 were proved infringed as no reasonable jury could have concluded that the three claim limitations disputed by Cisco are not met by the UCS.

1. No Reasonable Jury Could Conclude that the UCS Does Not Program CPUs to Establish Network Topology.

The centerpiece of Cisco's noninfringement defense was its assertion that the UCS does not program CPUs. But it was uncontested at trial that the CPUs in the UCS must be loaded with drivers and MAC addresses before the CPUs can communicate with other CPUs. While "program" was not formally construed, the loading of drivers and MAC addresses constitutes "programming" the CPUs under any reasonable understanding of that term. Further, because the CPUs cannot communicate until the drivers and MAC addresses are programmed—there is no network topology—until the CPUs are programmed. The UCS thus meets the "program said corresponding set of computer processors . . . to establish the specified virtual local area network topology" limitation of claim 3 and the "programming said corresponding set of computer processor[s] . . . to establish the specified virtual local area network topology" of claim 7.

The only reasoning the district court provided "with respect to the programming step, [was that] Cisco witnesses testified that the central processing unit (CPU) is *not* programmed for the claimed purpose of establishing the specified virtual local area topology – *only* the network interface card (NIC) is."

APPX00078 (emphasis in original). In other words, the district court implicitly acknowledged that the CPUs are in fact "programmed," but denied JMOL on the

grounds that the CPUs are allegedly not programmed “for the claimed purpose” of establishing network topology. But the district court’s reference to “the purpose” improperly injected a state of mind requirement into the claim language. *See Embrex, Inc. v. Serv. Eng’g Corp.*, 216 F.3d 1343, 1353 (Fed. Cir. 2000) (Rader, J., concurring) (“[T]he Supreme Court and this court have recently reiterated that intent is irrelevant to infringement.”) Indeed, the district court’s new gloss on the claim term “to establish the specified virtual local area network topology” amounted to an impermissible “reconstruction” of a claim term at the JMOL stage. *See Wi-LAN, Inc. v. Apple, Inc.*, 811 F.3d 455, 465 (Fed. Cir. 2016) (“It is too late at the JMOL stage to argue for or adopt a new and more detailed interpretation of the claim language and test the jury verdict by that new and more detailed interpretation.”) (internal quotations and brackets omitted).

Cisco’s argument that only the NIC is programmed to establish the network topology is too constrained and narrow. While establishing network topology involves programming the NIC, there is no network topology until the CPUs can communicate with each other, which does not occur until *the CPUs are programmed* with the drivers and MAC addresses. The “purpose” of a particular programming operation is irrelevant. What matters is (1) whether the CPUs are programmed (they are) and (2) the result of programming the CPUs (the network

topology is established). Egenera's expert was unambiguous and unrebutted on this point, testifying:

Q Now, counsel just asked you questions about the creation of the VNIC alone. If the VNIC is created but the CPU has not been configured or booted, is there a topology such that messages from the CPU can be communicated among a set of processors in the UCS?

A No. Until the CPU is booted and has the information, such as the MAC address and information about the VNIC, it can't communicate with the other processors in the topology.

APPX13207, 109:5-12.

No reasonable jury could have found noninfringement based on the "programming the CPUs" limitations.

2. No Reasonable Jury Could Conclude that the UCS Does Not Modify Messages to the External Communication Network.

Cisco's second noninfringement argument was that the UCS does not modify messages to the external communication network and thus does not "modify said received messages to transmit said modified messages to the external communication network" as required by claims 3 and 7. But this element, too, was shown to be present in the accused product by unrefuted evidence. Infringement as to this limitation hinges on the meaning of the unconstrued term "modify." The UCS Fabric Interconnect, which acts as the claimed "control node," receives messages from the CPUs and removes a VN-Tag from those messages before

sending them to the external communication network. Removing information from the messages is inarguably a modification of those messages because the messages are different after the VN-Tag is removed. The UCS Fabric Interconnect thus modifies messages before sending them to the external communication network.

The district court again declined to provide any reasoning for rejecting this evidence, instead summarily declaring that “Cisco witnesses similarly testified that the accused product does not practice the modifying or extracting elements for reasons tied directly to the claim language.” APPX00078. But Cisco’s witnesses provided no competent testimony refuting Egenera’s evidence.

In opposing Egenera’s JMOL motion, Cisco once again relied on irrelevant NIC operations in seeking to deflect from its infringing design. In particular, Cisco asserted that the VN-Tag is added to the message from the computer processor by the NIC, and thus the CPU message with the VN-Tag received by the Fabric Interconnect is not the same message *sent by the CPU*. APPX13764–13765. But this ignores the claim language, which requires only that the control node modify the “received” messages, *i.e.* the messages *as they are received*, not *as they are sent from the CPU*.

Further, even if the control node were required to modify the message sent from the CPU, adding and removing a VN-tag to a message are *both* modifications. Even under Cisco’s reading of the claims, then, the message from the CPU is

modified once by the NIC and again by the UCS Fabric Interconnect. The additional modification by the NIC does not detract from the modification by the UCS Fabric Interconnect, which meets the claim limitation. It is well-settled that an additional step does not defeat infringement of a “comprising” claim because “[i]nfringement arises when all of the steps of a claimed method are performed, whether or not the infringer also performs additional steps.” *Smith & Nephew, Inc. v. Ethicon, Inc.*, 276 F.3d 1304, 1311 (Fed. Cir. 2001); *see also Outside the Box Innovations, LLC v. Travel Caddy, Inc.*, 695 F.3d 1285, 1305 (Fed. Cir. 2012) (“The usage ‘comprising’ means that additional components may be present in the device, but does not change the elements that are stated in the claim.”).

Nothing in the claims requires that the content of the message received by the control node for modification be identical in all respects to the content of the message when it left the processor or transmitting messages directly from the processor to the control node. The modification in claim 3 is applied to “said *received* messages,” and thus what matters is the state of the messages when they are received by the control node. APPX00116, 29:58. Similarly, in claim 7 the messages must be “received and modified” by the control node, and thus modification is tied to receipt by the control node rather than transmission by the CPU. APPX00117, 32:35-36. Reading in a requirement that the messages received by the control node has been unchanged since transmission from the CPU would

be error. *See generally Genentech, Inc. v. Chiron Corp.*, 112 F.3d 495, 501 (Fed. Cir. 1997) (“To be joined or connected does not necessitate a *direct* joining or connection.”)).

Cisco also argued that Egenera had failed to demonstrate infringement under the means-plus-function construction of claim 3. APPX13767. But Cisco never argued noninfringement based on means-plus-function claiming to the jury, and thus the record is void of any suggestion that the jury could have reached its noninfringement finding on that basis. In any case, this argument would not support the jury’s noninfringement finding as to claim 7, which does not contain the “logic” language that was construed as a means-plus-function term.

The unrebutted evidence showed that the UCS Fabric Interconnect receives a message from the CPU and modifies it by removing VN-tag before sending it to the external communication network. Under any reasonable evaluation of the evidence, this evidence shows the accused product meets the disputed claim limitation, and the jury’s noninfringement finding cannot reasonably be supported on this basis.

3. No Reasonable Jury Could Conclude that the UCS Does Not Extract an Address and Identify a Corresponding Address for Messages to the External Storage Network.

Cisco’s only remaining noninfringement argument was that the UCS does not “extract an address from a received storage message, to identify the defined

corresponding address in the external storage address space, and to provide messages on the external storage network corresponding to the received storage messages and having the corresponding address” as required by claims 3 and 7.

This phrase, too, was not formally construed by the district court, but Egenera presented un rebutted evidence that the plain and ordinary meaning of this limitation was satisfied. To prove this claim element was met, Egenera showed that the UCS Fabric Interconnect performs a FLOGI to FDISC conversion process for messages to the external storage network. Egenera’s expert explained that the UCS Fabric Interconnect will extract a VLAN ID and identify the corresponding VSAN ID number for transmission of the message to the external storage network.

APPX12711–12712, 92:23–93:14. Dr. Jones explained that the Fabric Interconnect also will extract an address during the conversion of FLOGI to FDISC for messages sent to the external storage network:

[A]s we saw earlier in the structure for the FI [Fabric Interconnect], in examples up here, it extracts information including header information such as addresses. This is the storage message going out to the storage network. It will identify a corresponding address for which this message should be sent and it’s going to be sent out over a particular port, a port associated with that external storage network and will provide messages to that storage network over that particular external port.

APPX12712–12713, 93:15–94:1.

As with the claim limitations discussed *supra*, the district court did not explain its refusal to accept this evidence, instead stating that “Cisco witnesses similarly testified that the accused product does not practice the modifying or extracting elements for reasons tied directly to the claim language.” APPX00078. But, again, Cisco did no such thing. In opposing Egenera’s JMOL motion, Cisco asserted that “a message destined for the external storage network is given a ‘D_ID,’ or destination address, which does not change as the message travels through the UCS.” APPX13767. But the fact that other address information exists is not germane to Egenera’s proof that address information is extracted elsewhere in the process.

Cisco’s only response to Egenera’s actual infringement theory was to argue that “FLOGI and FDISC are not addresses.” APPX13768. But this misinterprets Egenera’s infringement position. As discussed above, Egenera did not assert that FLOGI and FDISC themselves are addresses, but that VLAN ID and VSAN ID are addresses, with the former being extracted and the latter corresponding to the former. This is all that the claim limitation requires.

Cisco offered no substantive rebuttal to this evidence in opposing Egenera’s JMOL motion other than to quibble over language. APPX13767–13769. First, Cisco cited testimony in which Egenera’s expert stated that the VLAN ID and VSAN ID “act as addresses,” and asserted that this suggests that they are not

themselves addresses. APPX13768–13769. But while Cisco characterizes these values as “tags,” there is no dispute that, as Dr. Jones explained, they “are used to route and deliver messages to the correct locations” and thus “act as addresses within the UCS.” APPX12676, 57:13-15. Thus, these “tags” would be considered “addresses” under any reasonable interpretation of that term.

Cisco also asserts that “no Cisco and Egenera documents discuss using a VLAN ID or a VSAN ID as a storage address.” APPX13769. Cisco’s reference to “Egenera documents” confirms that it was improperly comparing the accused product to the BladeFrame *product* rather than the patent itself, which cannot sustain a finding of noninfringement. *Zenith*, 19 F.3d at 1423 (“As we have repeatedly said, it is error for a court to compare in its infringement analysis the accused product or process with the patentee’s commercial embodiment or other version of the product or process; the only proper comparison is with the claims of the patent.”). In any case, there is no requirement that the accused product use the exact same terminology as the patent claims. As there is no dispute that the VLAN ID and VSAN ID act as addresses, the jury could not reasonably have concluded otherwise. To the extent the district court’s JMOL denial was based on the determination that acting as an address does not satisfy the “address” limitations, this ruling was based on an erroneous and belated claim construction. *Wi-LAN*, 811 F.3d at 465. At the very least, the VLAN ID and VSAN ID, by acting as addresses,

meet the “address” limitations under the doctrine of equivalents. *See, e.g., UCB, Inc. v. Watson Lab’ys. Inc.*, 927 F.3d 1272, 1284 (Fed. Cir. 2019) (affirming a finding of infringement under the doctrine of equivalents where the accused product “act[s] as” the claimed invention).

Because none of Cisco’s noninfringement arguments could reasonably support the jury’s verdict of noninfringement, the district court should have entered JMOL in favor of Egenera on infringement.

4. The Jury’s Verdict Cannot Be Sustained Based on Other Claim Limitations.

In opposing Egenera’s JMOL motion, Cisco vaguely argued that “[t]he jury could have found noninfringement based on any number of claim terms in addition to the three addressed above.” APPX13770. But the three noninfringement arguments discussed above were the only ones that Cisco presented to the jury in its summation. Although the jury returned a general verdict, Cisco cannot rely on the overall burden of proof of infringement to defeat JMOL when Cisco only disputed three claim elements at trial. For example, in *Snuba Int’l, Inc. v. Dolphin World, Inc.*, No. 99-1357, 2000 WL 961363, at *3-4 (Fed. Cir. July 11, 2000), the Court overturned a “general verdict” where “the only disputed issues at trial” were discrete claim elements and no reasonable jury could conclude that the disputed

elements were not met. Here, Cisco only disputed three claim elements, and all of them are met by the UCS.

5. The Court Should Vacate the District Court's Judgment and Remand for Further Proceedings.

In view of the evidence discussed above, the Court should vacate the district court's judgment and remand the case with instructions to (1) enter JMOL of infringement in favor of Egenera and (2) conduct a new trial on willfulness and damages.

D. Egenera is Entitled to a New Trial Based on the Weight of the Evidence and the District Court's Myriad Errors.

Should the Court decline to enter JMOL in Egenera's favor, the Court should remand with instructions to order a new trial.

1. Egenera's Unrebutted Evidence of Infringement Warrants a New Trial.

The weight of the evidence supporting Egenera and the lack of legally relevant evidence supporting the noninfringement verdict is at least sufficient to warrant a new trial. *See Jennings v. Jones*, 587 F.3d 430, 439 (1st Cir. 2009) ("In some cases, the evidence might preclude judgment as a matter of law and yet lean so heavily in the other direction so as to justify a district judge in ordering a new trial."). As explained above, Egenera presented substantial, unrebutted, evidence that the requirements of claims 3 and 7 of the '430 Patent were met by the accused

UCS. Cisco effectively conceded all but three of these requirements. But as noted above, for these three disputed elements, Cisco did not meaningfully dispute Egenera's evidence regarding the operation of UCS. Instead, Cisco sought to confuse the jury by drawing distinctions between its product and Egenera's BladeFrame, pointing to irrelevant unclaimed features presented in its product. A new trial is therefore warranted based on the weight of the evidence.

2. The District Court's Erroneous Jury Instruction Regarding "Copying" Warrants a New Trial.

An erroneous instruction to the jury warrants a new trial when it "can fairly be said to have prejudiced the objecting party." *Kennedy v. Town of Billerica*, 617 F.3d 520, 529 (1st Cir. 2010). That the error was made during empanelment rather than in the district court's final jury instructions is of little moment, as misstatements from a trial court during general orientation can require reversal of the verdict if "it produces prejudice or misleads the jury in a material way." *United States v. Hernandez*, 176 F.3d 719, 731 (3d Cir. 1999) (quoting *People v. Ignacio*, 852 F.2d 459, 461 (9th Cir. 1988)) (internal quotations omitted). Here, the district court's erroneous "copying" instruction, given at the onset of trial, undoubtedly prejudiced Egenera by influencing the jury to perceive and observe the evidence for signs of copying rather than comparing the accused system to the claims of the '430 Patent.

As discussed above, Egenera's evidence of infringement was clear and un rebutted. Rather than meet this evidence head-on, Cisco instead focused its evidence and arguments on the legally irrelevant question of whether the accused Cisco UCS *copied* Egenera's patented BladeFrame product. Each of Cisco's witnesses intertwined testimony directed to a purported lack of copying with Cisco's non-infringement defense.

For example, when testifying that the NIC, rather than the CPU, was programmed, Mr. Dvorkin testified that it was not a "copy" of Egenera's design. APPX12815, 42:14-20. Mr. Jayakrishnan, similarly testified that UCS was not a "copy" of Egenera immediately before testifying that UCS did not infringe the '430 Patent. APPX12891, 118:15-18. Dr. Jeffay, when giving testimony to "help this jury figure out whether or not Cisco infringes the ['430] patent," was asked to "start with copying" and concluded that he "saw no evidence of any copying," including no documents and "no physical evidence." APPX12910-12918, 137:22-145:4. During redirect, Dr. Jeffay again tied his infringement analysis to copying, stating that he was "retained to do an analysis of the UCS and compare it to the '430 patent to assess whether or not there was infringement of the '430 patent," followed immediately by his conclusion that there was no evidence of "copying" based on all of the evidence in the case. APPX13051-13052, 111:13-112:6.

During closing arguments, Cisco’s counsel repeatedly raised the issue of copying, mentioning copying over 30 times—more times than infringement—blurring the lines between the two concepts. *See, e.g.*, APPX14976, 10-38:6-10 (“We don’t have Giganet on the inside like they do, so we don’t need to do what they do to get to storage, and we don’t need to do what they do to get to the Internet. It’s totally different. We have our own patents on it. The idea that we copied is nonsense.”). Cisco’s intermingling of copying with infringement exacerbated the prejudice to Egenera caused by the district court’s erroneous instruction to the jury panel.

The district court could easily have at least mitigated its error with a curative jury instruction but inexplicably chose not to do so despite the clear legal principle that “when an erroneous [preliminary] instruction is given, a subsequent clarification must be sufficiently clear and compelling to allow a reviewing court to conclude that there was no reasonable likelihood that the initial inaccuracy affected the jury’s deliberations.” *Hernandez*, 176 F.3d at 731 (citing *Victor v. Nebraska*, 511 U.S. 1 (1994)). The district court’s failure to cure its error warrants a new trial.

3. The District Court's Refusal to Instruct the Jury that a Patented Product May Infringe Another Patent Warrants a New Trial.

The jury's apparent confusion regarding copying was further exacerbated by the district court's refusal to include Egenera's proposed jury instruction that a patented product can still infringe another patent. Without this instruction, Cisco was emboldened to repeatedly emphasize its own legally irrelevant patents as evidence that it did not infringe because it did not "copy" Egenera's patent. That argument was legally improper, nevertheless it was allowed by the district court, and as this Court has recognized, the repeated uncorrected use of that argument to the jury created a significant risk that the jury was confused into thinking the defendant's later-filed patents covering the accused product resulted in noninfringement. *See, e.g., Glaros v. H.H. Robertson Co.*, 797 F.2d 1564, 1572–73 (Fed. Cir. 1986) (affirming exclusion of patents covering accused product that would "create side issues" and "unduly distract[] the jury").

Egenera sought to forestall this confusion with a simple jury instruction: "A product that is covered by a subsequent or later patent may still infringe an earlier patent." APPX11456. The district court refused to provide this instruction, meaning Cisco was able to bring its own irrelevant patents into the case and mislead the jury as to their significance. Indeed, Cisco explicitly relied on *its own patents* to argue noninfringement, *e.g.* by asserting that its own patents do not

equate VSAN IDs with addresses. APPX12991, 51:6-16; *see also* APPX12990, 50:2-7. Similarly, when arguing that the UCS Fabric Interconnect did not modify messages received from the CPU, Dr. Jeffay relied on Cisco’s allegedly patented VN-tag technology as a basis for asserting that the messages received from the CPU and the messages leaving the Fabric Interconnect are the same. APPX13008–13010, 68:9–70:11.

Naturally, Cisco’s counsel relied heavily on this testimony during closing arguments, reiterating the connection between Cisco’s non-infringement defenses and Cisco patents covering UCS. For example, Cisco’s counsel reiterated Dr. Jeffay’s argument that UCS does not “have to modify anything going to storage” and thus does not infringe because its system is “an entirely separate invention” for which Cisco has a patent. APPX14975, 10-37:12-22 (“It’s an entirely separate way of doing things, and we’ve got a patent on it.”). Cisco’s counsel likewise relied on Cisco’s VN-tag patents, arguing that “[w]e are not using the Egenera way. We’ve got a patent on that too. We patented . . . our VN tagging. It’s entirely different from what they do.” *Id.* at 10-37:23-25. Cisco’s reliance on its own patents for its non-infringement defenses is affirmed by its Law360 interview immediately after the jury verdict, with Cisco “proud of . . . our proven track record of delivering innovation,” and confirming it had used the trial to “showcase Cisco’s patents on these innovative technologies.” APPX13754–13755.

The district court could have prevented, or at least mitigated, Cisco's improper reliance on its own irrelevant patents with a simple and sought jury instruction. It chose not to give that instruction, and a new trial is warranted free of the distraction of Cisco's irrelevant patents.

4. Cisco's Improper Closing Arguments Warrant a New Trial.

Cisco's reliance on irrelevant and emotionally-charged arguments extended further into its counsel's summation, during which Cisco's counsel attempted to sway the jury by implying that Egenera is a non-practicing entity and improperly highlighting "empty chairs" in violation of the district court's orders *in limine*.

Cisco's improper conduct warrants a new trial under the First Circuit's plain error review standard.³ *United States v. Carpenter*, 494 F.3d 13, 18–21 (1st Cir. 2007).

Plain error requires "error, plainness, prejudice, and miscarriage of justice or something akin to it." *Chestnut v. City of Lowell*, 305 F.3d 18, 20 (1st Cir. 2002).

There can be no reasonable dispute that Cisco's flouting of the district court's orders and other improprieties in its closing argument, detailed above, was error, and that the error was "plain." Cisco's improper arguments were so

³ The district court applied the plain error standard rather than the totality of the circumstances standard because it ruled that Egenera did not properly object to Cisco's counsel's closing arguments during trial.

prejudicial to Egenera that a new trial is needed to vindicate Egenera's right to a fair trial and prevent a miscarriage of justice.

Pejoratively referring to “venture capitalists” trying to “make good on a bad bet” leveraged negative perceptions of venture capitalists to improperly bias the jury against Egenera. Egenera was doubly prejudiced because it did not have the ability during closing to present evidence in rebuttal. *See Anheuser-Busch, Inc. v. Nat. Beverage Distribs.*, 69 F.3d 337, 347 (9th Cir. 1995) (“Because references to prospective purchasers and to the details of potential sales were excluded by the court’s ruling, Anheuser did not present evidence to rebut the inference Beardslee had created. Anheuser was thus in the untenable situation that the district court had sought to avoid in its *in limine* ruling.”). Further coupled with Cisco’s empty chair arguments and suggestion that Egenera is a non-practicing entity with no employees, there is little doubt that the jury entered deliberations with the false impression that Egenera’s lawsuit was an immoral, reckless money-grab. *See Polansky v. CNA Ins. Co.*, 852 F.2d 626, 630 (1st Cir. 1988) (condemning “thinly veiled yet successful attempt[s] to introduce an emotional element into the jury’s deliberations”); *Carpenter*, 494 F.3d at 24 (affirming grant of new trial where inflammatory use of gambling metaphors during closing arguments “was an appeal to instincts of moral disapprobation about recklessness, waste, and perhaps even theft”); *see also Commil USA, LLC v. Cisco Sys., Inc.*, 720 F.3d 1361, 1370 (Fed.

Cir. 2013), *vacated on other grounds by Commil USA, LLC v. Cisco Sys., Inc.*, 135 S. Ct. 1920 (2018) (affirming grant of new trial where “Cisco attempted to instill in the jury, through irrelevant references to ethnicity and religion, an ‘us versus them’ mentality”). Cisco skewed the verdict by appealing to juror bias. This Court’s interest in ensuring that the jury’s verdict here was not a miscarriage of justice warrants a new trial.

5. Cisco’s Improper Lay Opinion Testimony Warrants a New Trial.

Finally, Egenera is entitled to a new trial free of Cisco’s extensive improper lay opinion testimony. The district court’s denial of Egenera’s motion *in limine*, APPX00165, ECF 411, ¶ 3, and trial objections, APPX12776, 3:19-24, APPX12816, 43:2-7, opened the door for Mr. Dvorkin and Mr. Jayakrishnan to provide detailed non-infringement opinion testimony that “confuse[d] the jury into concluding that the testimony was relevant to the issue[] of infringement.” *Omega*, 920 F.3d at 1353. In effect, Mr. Dvorkin and Mr. Jayakrishnan became surprise expert witnesses for Cisco who did not provide expert reports, effectively precluding Egenera’s expert from testing their opinions and responding in kind. *See HVLPO2, LLC v. Oxygen Frog, LLC*, 949 F.3d 685, 690–91 (Fed. Cir. 2020) (ordering a new trial based on lay witness testimony on obviousness). Admission of their improper testimony “deprive[d] [Egenera] of its right to have the question

of [infringement] decided based on admissible, qualified expert testimony, [and] it prejudiced [Egenera] by not affording it the appropriate procedures for testing such testimony.” *Id.* at 690. A new trial is required because “[t]here is no way to know whether [their] improper testimony provided some or all of the basis for the jury’s decision.” *Id.*

6. The District Court Erred in Denying Egenera’s Motion for a New Trial.

Regarding the curative jury instruction, the district court sought to distinguish between “introductory remarks made to the venire during the winnowing down of prospective jurors with the formal instructions given to the actual jury once seated.” APPX00083. Because the district court’s incorrect “copying” comment was not conveyed as a formal instruction, the district court reasoned, there was no need to cure a clearly erroneous statement from the court via a further instruction. But courts have held that misstatements from the Court during general orientation can require reversal of the verdict if “it produces prejudice or misleads the jury in a material way.” *Hernandez*, 176 F.3d at 731 (quoting *Ignacio*, 852 F.3d at 461) (internal quotations omitted). The district court did not address this point, and adopting the district court’s reasoning would open trial proceedings to unlimited errors such as the one the district court made during empanelment. Indeed, under the district court’s logic, a trial judge could say

anything during “introductory remarks” and have no obligation to correct them, regardless of how erroneous or prejudicial the remarks.

As to the jury instruction concerning earlier patents, the district court dispensed of Egenera’s motion in a single paragraph, holding that Egenera’s proposed instruction was not necessary because:

Prior to the deliberations, the court instructed the jurors that (1) what mattered for infringement purposes was whether the product met all elements of the claims themselves (*i.e.*, not whether it was subject to any other patents); and (2) the presence of additional features would not defeat a showing of infringement.

APPX00082–00083. But the district court ignored the legal authority cited by Egenera holding that juries should not be confused or distracted by the defendant’s patents and failed to explain its refusal to prevent jury confusion with a simple one-sentence instruction from Egenera.

The district court also split hairs concerning Egenera’s concerns over Cisco’s closing argument. The district court adopted Cisco’s argument that Mr. Thompson in fact appeared at trial when he only appeared by deposition designation. APPX00080. The district court also let Cisco’s inflammatory insinuations about Egenera’s status as a practicing entity because Cisco did not “expressly” call Egenera a “non-practicing entity” or a “patent troll.” *Id.* And the district court dismissed any possibility of prejudice based, again, on its own formal

jury instruction. APPX00081. All of this ignored import of the district court's rulings *in limine* by allowing Cisco to skirt the spirit of the district court's rulings with carefully selected language. This was highly prejudicial to Egenera and warrants a new trial.

Finally, in a single paragraph, the district court dismissed Egenera's challenges to Cisco's improper lay opinion testimony because Egenera purportedly waived its objection by declining to object during examination. *Id.* This, again, was an overly pedantic analysis of the record by the district court. The law of this Court and of the First Circuit is clear that a denied pretrial motion *in limine* preserves all objections to the subject evidence during trial, and that objections to the evidence need not be raised repeatedly during trial. *See* Fed. R. Evid. 103(b) ("Once the court rules definitively on the record — either before or at trial — a party need not renew an objection or offer of proof to preserve a claim of error for appeal."); *Uniloc*, 632 F.3d at 1319 ("Microsoft's in limine filings" "made its position on this evidence sufficiently clear to preserve the instant challenge"); *Crowe v. Bolduc*, 334 F.3d 124, 133 (1st Cir. 2003) ("If . . . the in limine ruling is final and unconditional, the issue was preserved for appeal and no further steps need be taken to preserve the issue."). There is no dispute that Egenera moved *in limine* to exclude lay opinion testimony and there is no dispute that the district court denied that motion. APPX00165, ECF 411, ¶ 3.

Despite this record, the district court held that Egenera's objections to lay opinion testimony had been waived because, while the district court "denied Egenera's motion to exclude the opinions of these witnesses," the district court "did not authorize the witnesses to offer expert opinions." APPX00081, n.3. But this formalism cannot support a finding of waiver *in toto* to Cisco's extensive improper lay opinion testimony. The district court certainly cited no authority to the contrary.

The district court also held that there could be no prejudice simply because Egenera asked Mr. Jayakrishnan "about the '430 patent," and because "much of what Mr. Dvorkin attested to simply 'mirrored' the testimony of Mr. Jayakrishnan." APPX00082. But the district court failed to explain how the numerous improper statements by Cisco's lay witnesses were tied to Egenera's questioning, and thus the district court did not justify its finding of lack of prejudice.

Egenera *was* prejudiced by Cisco's lay opinion testimony. At a minimum, Cisco was permitted to offer evidence of litigation-driven test results that had never been disclosed to Egenera. This alone warrants a new trial on infringement.

VII. CONCLUSION AND STATEMENT OF RELIEF SOUGHT

Egenera respectfully requests that the Court:

- 1) Vacate the district court's ruling of summary judgment of noninfringement of claims 1 and 5 of the '430 Patent and remand for further proceedings; and
- 2) Vacate the district court's judgment and remand with instructions to enter JMOL of infringement of claims 1 and 5 and conduct a new trial on willfulness and damages or, in the alternative, vacate the district court's judgment and remand with instructions to conduct a new trial on all issues.

Respectfully submitted,

Date: April 26, 2023

By: /s/ Matthew C. Holohan
Robert R. Brunelli
rbrunelli@sheridanross.com
Matthew C. Holohan
mholohan@sheridanross.com
SHERIDAN ROSS P.C.
1560 Broadway, Suite 1200
Denver, Colorado 80202
Telephone: (303) 863-9700
Facsimile: (303) 863-0223
litigation@sheridanross.com

ADDENDUM

Date	Description	Bates Nos.
07-17-2017	Stipulation and Protective Order Regarding Protected Information	APPX00001-APPX00036
06-23-2021	Memorandum and Order on Cross Motions for Summary Judgment and to Exclude Expert Testimony	APPX00037-APPX00076
12-15-2022	Memorandum and Order on Motion for Judgment as a Matter of Law and Motion for a New Trial	APPX00077-APPX00084
12-16-2022	Judgment	APPX00085
	U.S. Patent No. 7,231,430	APPX00086-APPX00119

**IN THE UNITED STATES DISTRICT COURT
FOR THE DISTRICT OF MASSACHUSETTS**

EGENERA, INC.,

Plaintiff,

v.

CISCO SYSTEMS, INC.,

Defendant.

Civil Action No. 1:16-cv-11613-RGS

STIPULATION AND PROTECTIVE ORDER

REGARDING PROTECTED INFORMATION

TABLE OF CONTENTS

	Page
I. INFORMATION SUBJECT TO THIS ORDER.....	1
A. Information Designated “Confidential”	2
B. Information Designated “Confidential – Outside Counsel Only”	4
C. Information Designated “Confidential – Outside Counsel Only – Source Code”	5
D. Use of Protected Information at Trial	15
II. HOW TO DESIGNATE PROTECTED INFORMATION	15
III. PROSECUTION BAR.....	17
IV. DISCLOSURE OF TECHNICAL ADVISERS AND IN-HOUSE COUNSEL	18
V. CHALLENGES TO CONFIDENTIALITY DESIGNATIONS.....	21
VI. LIMITATIONS ON THE USE OF PROTECTED INFORMATION	22
VII. NON-PARTY USE OF THIS PROTECTIVE ORDER.....	25
VIII. NO WAIVER OF PRIVILEGE	25
IX. INADVERTENT DISCLOSURE NOT AUTHORIZED BY ORDER.....	26
X. MISCELLANEOUS PROVISIONS.....	27

Upon motion of the above-named parties that the Court enter this Stipulation and [Proposed] Protective Order Regarding Protected Information pursuant to Fed. R. Civ. P. 26(c), it is hereby ORDERED THAT:

I. INFORMATION SUBJECT TO THIS ORDER

1. All documents, materials, items, and information produced either by a Party or a non-Party (each such Party and non-Party is referred to individually as a “Producing Party”) to any of the Parties in this action (a “Receiving Party”) shall be governed by this Protective Order (“Order”).

2. Discovery material produced in this action may be labeled as one of three categories: “Confidential,” “Confidential – Outside Counsel Only,” or “Confidential – Outside Counsel Only – Source Code,” as set forth in Sections I.A through C below. All three categories of information shall be referred to collectively in this Order as “Protected Information.” This Order protects discovery material, any information copied or extracted from discovery material, and all copies, excerpts, summaries, or compilations of such information, as well as testimony, conversations, or presentations by Parties or their counsel in court or in other settings that might reveal Protected Information.

3. Any document or tangible thing containing or including any Protected Information may be designated as such by the Producing Party by marking it “Confidential,” “Confidential – Outside Counsel Only,” or “Confidential – Outside Counsel Only – Source Code” prior to or at the time copies are furnished to the Receiving Party, as set forth in Section II below.

4. The following information is not Protected Information:

(a) Any information that is or, after its disclosure to a Receiving Party, becomes part of the public domain as a result of publication not involving a violation of this Order or other obligation to maintain the confidentiality of such information;

(b) Any information that the Receiving Party can show was already publicly known prior to the disclosure; and,

(c) Any information that the Receiving Party can show by written records was received by it from a source who obtained the information lawfully and under no obligation of confidentiality to the Producing Party.

5. Any Protected Information obtained by any Party, including its outside counsel, consultants, and experts, from any person pursuant to discovery in this action may be used only for purposes of this action.

6. Nothing in this Order shall restrict in any way a Producing Party's use or disclosure of its own Protected Information.

A. Information Designated "Confidential"

7. For purposes of this Order, "Confidential" information shall mean all information or material produced for or disclosed in connection with this action to a Receiving Party that a Producing Party believes to contain or comprise confidential technical, sales, marketing, financial, or other sensitive information qualifying for protection under standards developed pursuant to Fed. R. Civ. P. 26(c), whether embodied in physical objects, documents, or the factual knowledge of persons, and which has been so designated by the Producing Party.

8. Documents designated "Confidential" and information contained therein shall be available only to:

(a) Outside counsel of record for the Parties to this action, including any attorneys, paralegals, technology specialists, and clerical employees of their respective law firms;

(b) Technical advisers and their necessary support personnel who have signed the form attached hereto as Attachment A; the term “technical adviser” shall mean independent outside expert witnesses or consultants (*i.e.*, not current or former employees of a Party, nor at the time of retention anticipated to become an employee of a Party) with whom counsel may deem it necessary to consult, and for whom no unresolved objections to such disclosure exist after proper notice has been given to all Parties;

(c) Up to two in-house counsel, who are members of at least one state bar in good standing, are responsible for managing this action, have signed the form attached hereto as Attachment A, and presently are not directly involved in patent prosecution activities or other competitive decision-making (such as decisions related to sales or marketing of commercial products, acquisition of technology or businesses, and licensing technology or intellectual property). In-house counsel may only view Protected Information designated “Confidential” either:

(1) When such “Confidential” information is made exhibit to, referred to, or relied upon within any motion, brief, or other paper filed with the Court or served by the Producing Party; or

(2) In the presence of outside counsel of record at outside counsel’s office and shall not be permitted to make copies, notes, summaries, abstracts, compilations, or any other records of such “Confidential” information.

(d) The Court, its technical advisor (if one is appointed), court personnel, jury, and court reporters or videographers recording testimony or other pretrial proceedings in this action;

(e) Any arbitrator or mediator designated by the Court or agreed to by the Parties;

(f) Independent legal translators retained to translate in connection with this action; independent stenographic reporters and videographers retained to record and transcribe testimony in connection with this action;

(g) graphics, translation, or design services retained by counsel for purposes of preparing demonstrative or other exhibits for deposition, and hearings, or other court proceedings in this action;

(h) non-technical jury or trial consulting services;

(i) mock jurors, provided they are only exposed to “Confidential” information during the course of a mock trial or similar exercise and separately sign onto this Order (by executing Attachment A) and agree to be bound to its terms prior to said mock trial or similar exercise. Mock jurors signing onto this Order need not be disclosed to the Producing Party; and

(j) Any other person with the prior written consent of the Producing Party.

B. Information Designated “Confidential – Outside Counsel Only”

9. The “Confidential – Outside Counsel Only” designation is reserved for confidential information that constitutes (a) commercially sensitive marketing, financial, sales, research and development, or technical data or information; (b) confidential information obtained from a non-Party, including, without limitation, customer- and user-identifying information, and information obtained from a non-Party pursuant to a current Non-disclosure Agreement (“NDA”); and (c) information or data relating to strategic plans. The following information (if non-public) presumably merit “Confidential – Outside Counsel Only” designation: trade secrets, pricing information, financial data, sales information, sales or marketing forecasts or plans,

business plans, product development information, engineering documents, testing documents, employee information, and other information of similar competitive and business sensitivity.

10. In determining whether information should be designated as “Confidential – Outside Counsel Only,” each Party agrees to use such designation only in good faith.

11. Documents designated “Confidential – Outside Counsel Only” and information contained therein shall be available only to the persons and entities listed in Paragraph 8(a)–(b), (d)–(j) subject to any terms set forth or incorporated therein and not to persons identified in Paragraph 8(c), absent the written consent of the Producing Party.

C. Information Designated “Confidential – Outside Counsel Only – Source Code”

12. Nothing in this Order shall be construed as an admission that Source Code is properly discoverable in this action or as an obligation for any Party to produce any Source Code in this action or any other case. The “Confidential – Outside Counsel Only – Source Code” designation is reserved for Confidential Information that contains or substantively relates to Source Code.

13. “Source Code” shall mean computer code, scripts, assembly object code, source code listing, comments for source code, source code revision histories, object code listings, comments for object code, object code revision histories, Hardware Description Language (HDL) or Register Transfer Level (RTL) files that describe the hardware design of ASIC or other chip, other electronic files used in network operations, and network operation revision histories. Software source code files include without limitation files containing Source Code in C, C+, C++, BREW, Java ME, J2ME, assembler, digital signal processor (DSP) programming languages, and other human-readable programming languages. Software source code files further include “include” files, “make” files, “link” files, and other human-readable text files used in the

generation or building of software directly executed on a microprocessor, micro-controller, or DSP.

14. All such Source Code, and any other Protected Information designated as “Confidential – Outside Counsel Only – Source Code,” shall be subject to the following provisions:

(a) Source Code shall be made available for inspection electronically in a text-searchable form in a directory structure that mirrors the directory structure of each set of the Source Code as maintained by the Producing Party (*e.g.*, using TAR files). The Producing Party shall provide a manifest of the contents of the Source Code Computers. This manifest, which will be supplied in electronic form (and may comprise multiple files), will list the name, location, and either the MD5 checksum or file length of every source and executable file in each set of Source Code on the computers. The Reviewing Party may print a copy of any portion of the manifest in the room containing the Source Code Computers. The manifest shall be treated as “Confidential – Outside Counsel Only – Source Code.”

(b) Source Code shall only be made available for inspection—not produced, except as provided for below—and shall be made available at the offices of the Producing Party’s primary outside counsel of record in this action. Such location shall be in the continental United States.

(c) Source Code will be loaded on up to two non-networked computers running the Windows 10 operating system (the “Source Code Computers”) that have read-only access, that are password protected, are maintained in a secure area, and have all ports, software, and other avenues that could be used to copy or transfer data blocked. No recordable media or recording devices of any kind, including without limitation cameras, cellular telephones, CDs,

DVDs, and disk drives, may be permitted into the room containing the Source Code Computers (other than the non-networked computer that the Producing Party will provide for purposes of typing notes). The Producing Party shall provide a secure location near the room for inspecting individuals to store such media or recording devices (*e.g.*, their cellular telephones).

(d) The Source Code Computers will be made available for inspection during regular business hours, upon reasonable notice to the Producing Party, which shall not be less than twenty business days in advance of the first requested inspection and three business days in advance of each subsequent requested inspection. For purposes of this Paragraph, “regular business hours” are defined as Monday through Friday, 9:30 AM to 5:30 PM local time, excluding holidays. Moreover, upon reasonable notice from the Receiving Party, the Producing Party shall make reasonable efforts to accommodate the Receiving Party’s request for access to the Source Code Computers outside of regular business hours. In each such notice requesting Source Code inspection, the Receiving Party shall list the individuals (including attorneys) seeking access to the Source Code. The Producing Party shall have the right to object to such access, and if an objection to any specific individual is made, that individual shall not have access to the Source Code until resolution of the objection. A Producing Party may object to an individual only the first time the Receiving Party identifies such individual for access to the Source Code, unless additional relevant facts arise after the first time that the Receiving Party identifies such an individual.

(e) The Source Code Computers shall include the following software utilities which will allow the Receiving Party to view, search, and analyze the Source Code: TextPad, Notepad, NotePad++, WordPad, Cygwin, Singular Emacs, Quick View Plus 10, MS Office 2003 Viewers (Word, Excel, PowerPoint), Adobe Acrobat, GVim, Eclipse, 7zip, PowerShell (with

scripting enabled), dtSearch, Python, Visual SlickEdit (current version), and WinRAR. The Parties agree to accommodate reasonable requests for additional search and review tools upon a good-faith showing that such tools are reasonably required.

(f) The Receiving Party's outside counsel and/or expert shall be entitled to take notes relating to the Source Code. Such notes shall not be used as an end-run around the limits on printing Source Code set forth in this Order. The Receiving Party's outside counsel and/or expert may copy terms used in the Source Code into the notes only to the extent reasonably necessary to support or rebut the claims and defenses of the Parties in this case. The Receiving Party's outside counsel and/or expert may not copy complete code modules or lines of the Source Code into the notes. If the notes contain any Source Code, then the notes will be treated and marked as "Confidential – Outside Counsel Only – Source Code." Otherwise, such notes shall be considered and marked as "Confidential – Outside Counsel Only." To the extent that the Receiving Party's outside counsel and/or expert requests to take notes electronically, rather than by handwriting, the Producing Party shall provide a non-networked laptop computer in the room in which a Source Code Computer is located, which the Receiving Party's outside counsel and/or expert may use to type their notes. The laptop will be connected to a non-networked printer and shall include both Notepad++ and sdelete software utilities. The Receiving Party's outside counsel and/or expert may print a copy of their notes at the end of each day of review only on paper pre-marked with the appropriate confidentiality designation, which will be provided by the Producing Party. Such notes shall be considered the work product of the Receiving Party; and the Producing Party shall make no attempt to retrieve the Receiving Party's notes from the non-networked computer. The Receiving Party's expert may scan the printed notes into electronic files so long as the notes do not contain Source Code. Such scanning may be

performed only on a non-networked scanner (to be provided by the Producing Party) connected to the non-networked laptop computer in the room in which a Source Code Computer is located. Such scanned copy of the notes may be copied from the non-networked laptop onto an encrypted flash drive (the Producing Party will provide two such flash drives, which may be reused throughout the action), which the reviewer may take with them at the end of the day. Any notes generated from the Source Code review may be shared with outside counsel and with any outside experts or consultants who are approved to view material designated “Confidential – Outside Counsel Only – Source Code” under paragraph 14(m)(2) of this Order. One paper copy of the reviewer’s notes may be made for outside counsel, and one additional paper copy may be made for each authorized outside expert or consultant (up to a maximum of 5 copies total for all experts and consultants). No electronic “soft” copies of the reviewer’s notes may be made except as otherwise permitted under this Order. Notice shall be provided to the Producing Party when any copy of a reviewer’s notes is created or transmitted under this paragraph. Except as permitted above, no copies of all or any portion of Source Code may leave the room in which the Source Code is inspected except as otherwise provided herein. Further, no other written or electronic record of Source Code is permitted except as otherwise provided herein. Finally, there shall be no wholesale copying of Source Code.

(g) No person shall copy, email, transmit, upload, download, print, photograph, or otherwise duplicate any portion of the designated Source Code, except that the Receiving Party may request a reasonable number of pages (no more than 30 contiguous pages at a time, and no more than 1000 pages total in this action, absent a showing of good cause) of Source Code to be printed by the Producing Party during a Source Code inspection, but only to the extent necessary for use in this action. Unless objecting as outlined below, the Producing

Party will send via next-day delivery the requested material on paper bearing Bates numbers and the legend “Confidential – Outside Counsel Only – Source Code” within seven business days. The Receiving Party’s Outside Counsel may make no more than three additional paper copies of any printed portions of the Source Code, not including copies attached to court filings or used at depositions, and shall maintain a log of all paper copies of the Source Code. The log shall include the custodian of each copy of Source Code, the name of all persons accessing each copy, and the date and time of each access. Within one business day of notice, the Receiving Party shall provide a copy of this log to the Producing Party. Printouts (including any copies) must be kept in a secure container at the offices of outside counsel of record or at the office of an expert approved under this Order. Paper copies may not be converted into electronic format (including for emailing) EXCEPT as needed for final versions of expert reports and the filing and service of papers, motions, and pleadings made under seal. Draft expert reports that contain any Source Code shall be treated as “Confidential – Outside Counsel Only – Source Code” and are subject to all of the restrictions on Source Code in this Order.

(h) If the Producing Party objects that the printed portions are not reasonably necessary to any case preparation activity, the Producing Party shall make such objection known to the Receiving Party within seven business days of the request by the Receiving Party. If after meeting and conferring the Producing Party and the Receiving Party cannot resolve the objection, the Producing Party shall be entitled, but not required, to seek a Court resolution of whether the printed Source Code requested is reasonably necessary to any case preparation activity. Contested Source Code printouts need not be produced to the Receiving Party until the matter is resolved by the Court. The failure by the Producing Party to move the Court for relief

within five business days after the conclusion of the meet and confer pursuant to this Paragraph shall constitute a waiver of that objection.

(i) Any printed pages of Source Code, and any other documents or things reflecting Source Code that have been designated by the Producing Party as “Confidential – Outside Counsel Only – Source Code,” may not be copied, digitally imaged, or otherwise duplicated, except in limited excerpts necessary to use or attach as exhibits to depositions, expert reports, or court filings as discussed below. Printed copies of the Source Code shall not be mailed except with prior notice to the Producing Party. Any copies of Source Code mailed pursuant to the foregoing shall be mailed with a tracking number and require a signature by the recipient.

(j) A list of names of persons other than outside counsel of record who will inspect the Source Code will be provided to the Producing Party in conjunction with any written (including email) notice requesting inspection. Each time an individual accesses a Source Code Computer, he or she shall sign a log, provided by the Producing Party, recording the name of the individual, the date, the time in, the time out, and whether any printed portions of the Source Code were requested. The Producing Party shall be entitled to have a person observe all entrances and exits from the Source Code inspection room and shall maintain a copy of the log. The Producing Party may visually monitor the activities of the Receiving Party’s representatives during Source Code review only to ensure that no unauthorized electronic records of the Source Code are created or transmitted in any way. The Producing Party may not videotape or electronically record the activities of the Receiving Party’s representatives during Source Code review.

(k) Unless otherwise agreed in advance by the Parties in writing, following each inspection, the Receiving Party's outside counsel and/or experts shall remove all notes, documents, and other materials from the room that may contain work product or attorney-client privileged information. The Producing Party shall return any found items left in the room following each inspection session.

(l) Printed copies of Source Code in the possession of the Receiving Party shall be maintained in a secured, locked area in a manner that prevents duplication of or unauthorized access to the Source Code. The Receiving Party will not copy, remove, or otherwise transfer any portion of the Source Code from the Source Code Computers including without limitation copying, removing, or transferring any portion of the Source Code onto any other computers or peripheral equipment. The Receiving Party will not transmit any portion of the Source Code in any way from the location of the Source Code inspection. Paper copies of Source Code may not be removed from the secured container in public places (*e.g.*, airports or airplanes). Paper copies of Source Code may not themselves be copied by the Receiving Party, except that paper copies of the Source Code may be copied and used as exhibits for a deposition, expert report, motion, or trial, provided that such copies are kept within a secure container during transport to and from deposition, trial, or service.

(m) Only the following individuals shall have access to "Confidential – Outside Counsel Only – Source Code" materials, absent the express written consent of the Producing Party or further Court order:

(1) Outside counsel of record for the Parties to this action, including any attorneys, paralegals, technology specialists, and clerical employees of their respective law firms;

(2) Up to four outside experts or consultants per plaintiff and defendant in each case and who are pre-approved in accordance with Paragraph 8(b) and specifically identified as eligible to access Source Code;

(3) The Court, its technical advisor (if one is appointed), court personnel, and court reporters or videographers recording testimony or other pretrial proceedings in this action. Court reporters and videographers shall not retain any portions of Source Code. If used during a deposition, the deposition record will identify Source Code exhibits by production numbers;

(n) The Receiving Party's outside counsel shall maintain a log of all copies of the Source Code (received from a Producing Party) that are made available by the Receiving Party to any qualified person under Paragraph 14(m) above. The log shall include the names of the recipients and reviewers of copies of Source Code and locations where the copies are stored. Within one business day of notice, the Receiving Party shall provide a copy of this log to the Producing Party. Upon request by the Producing Party, the Receiving Party shall provide reasonable assurances and/or descriptions of the security measures employed by the Receiving Party and by any qualified person that receives a copy of any portion of the Source Code;

(o) Except as provided in Section I.C of this Order, the Receiving Party may not create electronic images, or any other images, of the Source Code from the paper copy for use on a computer (*e.g.*, may not scan the Source Code to a PDF or photograph the code). The Receiving Party may create an electronic copy or image of limited excerpts of Source Code only to the extent necessary in a pleading, exhibit, expert report, discovery document, deposition transcript, other Court document, or any drafts of these documents ("Source Code Documents"). If a Receiving Party reasonably believes that it needs to submit or present a substantial portion

(*e.g.*, a complete function or complete file) of Source Code as part of a filing with, or hearing before, the Court, the Parties shall meet and confer as to how to make such a submission while protecting the confidentiality of the Source Code. If the Producing Party agrees to produce an electronic copy of all or a portion of its Source Code or provide written permission to the Receiving Party to make such an electronic copy, the Receiving Party's communication or disclosure of electronic files or other materials containing any portion of Source Code shall at all times be limited solely to individuals who are expressly authorized to view Source Code under the provisions of this Order. The Receiving Party shall only include such excerpts as are reasonably necessary for the purposes for which such excerpts of Source Code are used. Images or copies of Source Code shall not be included in correspondence between the Parties (references to production numbers shall be used instead) and shall be omitted from pleadings and other papers except to the extent permitted herein. All electronic copies must be labeled "Confidential – Outside Counsel Only – Source Code." Where the Producing Party has provided the express written permission required under this Paragraph for a Receiving Party to create electronic copies of Source Code, the Receiving Party shall maintain a log of all such electronic copies of any portion of Source Code in its possession or in the possession of retained experts or consultants, including the names of all reviewers and recipients of such electronic copies and the locations where the electronic copies are stored.

(p) To the extent portions of Source Code are quoted in another document, either (1) the entire document will be stamped and treated as "Confidential – Outside Counsel Only – Source Code" or (2) those pages containing quoted Source Code will be separately bound and stamped and treated as "Confidential – Outside Counsel Only – Source Code."

(q) All copies of any portion of the Source Code in whatever form shall be returned to the Producing Party or securely destroyed if they are no longer in use. Copies of Source Code that are marked as deposition exhibits shall not be provided to the court reporter or attached to deposition transcripts; rather, the deposition record will identify the exhibit by its production numbers.

(r) The Receiving Party's outside counsel may only disclose a copy of the Source Code to individuals specified in Paragraph 14(m) above (*e.g.*, Source Code may not be disclosed to in-house counsel).

D. Use of Protected Information at Trial

15. The Parties shall meet and confer prior to trial to discuss procedures for maintaining the confidentiality of Protected Information during the course of trial including the manner in which Source Code will be made available for inspection and used at trial.

II. HOW TO DESIGNATE PROTECTED INFORMATION

16. The Parties acknowledge the importance of client access to information necessary to client decision-making in the prosecution or defense of litigation and, therefore, agree that designations of information as Protected Information and responses to requests to permit further disclosure of Protected Information shall be made in good faith and not to impose burden or delay on a Receiving Party or for tactical or other advantage in litigation. Such designations shall be made according to the remaining paragraphs in this Section.

17. **Written discovery and documents and tangible things.** Written discovery, documents (including electronically stored information as that term is used in Federal Rule of Civil Procedure 34), and tangible things may be designated Protected Information by placing the appropriate designation on every page of the written material prior to production. For digital files, the Producing Party may: (1) mark each viewable page or image with the appropriate

designation; (2) include the designation in the electronic file name; or (3) mark the medium, container, or communication in which the digital files are contained. In the event that original documents are produced for inspection, the original documents shall be presumed “Confidential – Outside Counsel Only” during the inspection and re-designated as appropriate during the copying process.

18. **Depositions and testimony.** Parties or testifying persons or entities may designate depositions and other testimony with the appropriate designation by indicating such designation on the record at the time the testimony is given. Alternatively, any Party may designate testimony or information disclosed at a deposition by notifying all Parties in writing within ten days after the Party’s receipt of the final transcript of the specific pages and lines of the transcript that contain Protected Information. Until such ten-day period has passed, all information disclosed during a deposition not designated otherwise shall be deemed “Confidential – Outside Counsel Only.” Any Protected Information used in the taking of a deposition, and the portions of the deposition transcript dealing with such Protected Information, shall remain subject to this Order. In such cases, the court reporter shall be informed of this Order and shall be required to operate in a manner consistent with this Order. In the event the deposition is videotaped, the original and all copies of the videotape shall be marked by the video technician to indicate that the contents of the video tape are subject to this Order. Counsel for any Party shall have the right to exclude from oral depositions any person (other than the deponent, outside counsel for any Party, the court reporter, and videographer) who is not authorized by this Order to receive or access Protected Information.

19. **Intangible, non-testimonial material.** In the case of non-testimonial Protected Information not reduced to documentary or tangible form or which cannot be conveniently

designated as described above, such information may be designated “Confidential,” “Confidential – Outside Counsel Only,” or “Confidential – Outside Counsel Only – Source Code” by informing the Receiving Party of the designation in writing at the time of transfer of such information.

III. PROSECUTION BAR

20. Any person who has reviewed any of another Party’s or non-Party’s Protected Information related to confidential, technical aspects of products, information related to new inventions, technology under development, or Source Code shall not, for a period commencing upon first receipt of such information and ending two years following the conclusion of this action (including any appeals), engage in any Prosecution Activity (as defined below) on behalf of a Party asserting a patent in this action, its successor-in-interest, or a related entity or affiliate or engage in any Prosecution Activity involving claims on a method, apparatus, or system relating to virtual networks, including configuration, deployment, and fault-handling of such networks, the routing or processing of server requests for web content, or the technology embodied within the Patents-in-suit.¹

21. Prosecution Activity shall mean obtaining disclosure materials for new inventions and inventions under development, investigating prior art relating to those inventions, making strategic decisions on the type and scope of patent protection that might be available or worth pursuing for such inventions (or any others), writing, reviewing, advising on, consulting on, preparing, prosecuting, drafting, editing, amending, or approving new applications, continuations, divisionals, or continuations-in-part of applications to cover those inventions (or

¹ The patents currently asserted are U.S. Patents Nos. 6,971,044 and 7,231,430. “Patents-in-suit” encompasses any patents asserted at any time in this action and any related patents, patent applications, provisional patent applications, and divisionals.

any others), or strategically amending or surrendering claim scope during prosecution. Prosecution Activity does not include participation in any reexamination, covered business method review, *inter partes* review, or other post-grant review proceeding, except that any attorney representing Plaintiff who is not screened from accessing, reviewing, and being informed about the content of (or who has accessed, reviewed, or been informed about the content of) Defendant's Protected Information shall not draft or assist in the drafting of any claim or amendment to any claim of the Patents-in-suit for a period of one year after the resolution (including appeals) of this action. Attorneys representing Plaintiff who have not accessed, reviewed, or been informed about the content of Defendant's Protected Information and who are screened from accessing, reviewing, and being informed about the content of Defendants' Protected Information may draft or assist in the drafting of any claim or amendment to any claim of the Patents-in-suit. Nothing in Section III of this Order shall prevent any attorney from sending non-confidential prior art or other related non-confidential materials to an attorney involved in patent prosecution for purposes of ensuring that such prior art is submitted to the U.S. Patent and Trademark Office (or any similar agency of a foreign government) to assist a patent applicant in complying with its duty of candor. Nothing in this provision shall prohibit any attorney of record in this action from discussing any aspect of this action that is reasonably necessary for the prosecution or defense of any claim or counterclaim in this action with their client. Nothing in this provision shall prohibit any attorney of record in this action from assisting their client in reexamination proceedings, *inter partes* review or other post-grant reviews, subject to the restrictions in this Order.

IV. DISCLOSURE OF TECHNICAL ADVISERS AND IN-HOUSE COUNSEL

22. Information designated by the Producing Party under any category of Protected Information and such copies of this information as are reasonably necessary for maintaining,

defending, or evaluating this action may be furnished and disclosed to (subject to the limitations of Section I of this Order) the Receiving Party's in-house counsel or technical advisers and their necessary support personnel.

23. No disclosure of Protected Information to in-house counsel, technical advisers, or their necessary support personnel shall occur until that person has signed the form attached hereto as Attachment A, a signed copy has been provided to all Parties, and, to the extent there has been an objection under this Section, that objection is resolved according to the procedures set forth below.

24. A Party desiring to disclose Protected Information to a technical adviser or in-house counsel of the Party shall give prior written notice of the intended disclosure by email to all counsel of record in the litigation, and the Producing Party shall have seven business days after such notice is given to object in writing to the disclosure. No Protected Information shall be disclosed to such technical adviser(s) or in-house counsel until after the expiration of the foregoing notice period and resolution of any objection. The Party desiring to disclose Protected Information to a technical adviser must provide the following information for each technical adviser: name, address, *curriculum vitae*, current employer, employment history for the past ten years, and a listing of cases in which the technical adviser has testified as an expert witness at trial or by deposition within the preceding five years. The Party seeking to disclose Protected Information shall provide such other information regarding the person's professional activities as reasonably requested by the Producing Party to enable the Producing Party to evaluate whether good cause exists to object to the disclosure of Protected Information to the person.

25. Within seven days of such notice of intended disclosure, a Party objecting to disclosure of Protected Information to a technical adviser or in-house counsel shall state in

writing with particularity the ground(s) of the objection, if any, else the objection is waived. The objecting Party's consent to the disclosure of Protected Information to a technical adviser or in-house counsel shall not be unreasonably withheld, and its objection must be based on that Party's good-faith belief that disclosure of its Protected Information to the technical adviser or in-house counsel is likely to result in specific business or economic harm to that Party.

26. If after consideration of the objection, the Party desiring to disclose the Protected Information to a technical adviser or in-house counsel refuses to withdraw the request to disclose the Protected Information to the technical adviser or in-house counsel, that Party shall provide notice to the objecting Party. Within seven days of that notice, the Parties shall meet and confer regarding the objection. Thereafter, the objecting Party shall move the Court, within five business days of such meet and confer, for a ruling on its objection. The objecting Party shall have the burden of showing to the Court "good cause" for preventing the disclosure of its Protected Information to the technical adviser or in-house counsel. A failure to file a motion within the five business day period, absent an agreement of the Parties to the contrary, shall operate as an approval of disclosure of Protected Information to the technical adviser or in-house counsel. The Parties agree to cooperate in good faith to shorten the time frames set forth in Section IV of this Order if necessary to abide by any discovery or briefing schedules.

27. An initial failure to object to disclosure to a person under the preceding Paragraphs in this Section shall not preclude the Producing Party from later objecting to continued access by that person for actual violations of this Order. Resolution of any such objection shall follow the procedure outlined in the preceding Paragraphs in this Section. If relief is sought from the Court regarding such objection, no further disclosure to which the Producing Party objected shall be made prior to the Court's resolution of the objection; however,

the designated person may continue to access Protected Information that had already been disclosed prior to the date of the objection.

V. CHALLENGES TO CONFIDENTIALITY DESIGNATIONS

28. The Parties shall use reasonable care when designating documents or information as Protected Information. Nothing in this Order shall prevent a Receiving Party from contending that any documents or information designated as Protected Information have been improperly designated. A Receiving Party may at any time request that the Producing Party cancel or modify the Protected Information designation with respect to any document or information contained therein.

29. A Party shall not be obligated to challenge the propriety of a designation of any category of Protected Information at the time of production, and a failure to do so shall not preclude a subsequent challenge thereto. Such a challenge shall be written, shall be served on outside counsel for the Producing Party, and shall particularly identify the documents or information that the Receiving Party contends should be differently designated. The Parties shall use their best efforts to resolve promptly and informally such disputes. If an agreement cannot be reached, and only after the Parties have met and conferred regarding the dispute, the Receiving Party shall have ten days from the conclusion of the meet and confer to file a motion to compel the Producing Party to re-designate the documents or information in dispute. If the Receiving Party does not timely file a motion for re-designation, then the Protected Information in dispute shall remain subject to the designation made by the Producing Party. All Protected Information is entitled to confidential treatment pursuant to the terms of this Order until and unless the Parties formally agree in writing to the contrary or a contrary determination is made by the Court as to whether all or a portion of a Protected Information is entitled to confidential treatment at a particular designation.

VI. LIMITATIONS ON THE USE OF PROTECTED INFORMATION

30. Unless otherwise ordered by the Court, or agreed to in writing by the Producing Party, all Protected Information shall be held in confidence by each person to whom it is disclosed, shall be used only for purposes of this action, shall not be used for any business purpose or in connection with any other actual or contemplated legal proceeding, and shall not be disclosed to any person who is not entitled to receive such information as herein provided. All produced Protected Information shall be carefully maintained so as to preclude access by persons who are not entitled to receive such information.

31. Except as may be otherwise ordered by the Court, any person may be examined as a witness at depositions, hearings, and trial and may testify concerning all Protected Information of which such person has prior knowledge. A present director, officer, or employee of a Producing Party may be examined at deposition and may testify concerning all Protected Information produced by that Party. A former director, officer, or employee of a Producing Party may be examined at deposition and may testify concerning all Protected Information produced by that Party that appears on its face or is established by other documents or testimony to have been previously received from or communicated to that person and of which he or she has prior knowledge, including Protected Information that relates to matters on which the witness has personal knowledge and that has been produced by that Party. Non-Parties may be examined during deposition or testify concerning any Protected Information of a Producing Party that appears on its face or from other documents or testimony to have been received from or communicated to that non-Party as a result of any contact or relationship with the Producing Party or a representative of the Producing Party.

32. Nothing contained herein shall be construed to prejudice any Party's right to use any Protected Information in questioning a witness at any deposition or hearing provided that the

Protected Information is only disclosed to persons: (1) eligible to have access to the Protected Information under the terms of this Order; (2) eligible to have access to the Protected Information by virtue of his or her employment with the Producing Party; (3) whose name appears on the Protected Information, including being identified in the Protected Information as an author, addressee, or copy recipient of such information. Any person not qualified to view Protected Information under this Order shall be excluded from the portion of the examination that concerns such Protected Information, unless the Producing Party consents in writing or on the record to his or her presence.

33. All transcripts of depositions, exhibits, answers to interrogatories, pleadings, briefs, and other documents submitted to the Court that have been designated as Protected Information, or which contain information so designated, shall be filed under seal in compliance with Local Rule 7.2 and any other rules or orders by the Court governing such filings.

34. Outside counsel of record for the Parties are hereby authorized to be the persons who may retrieve confidential exhibits and other confidential matters filed with the Court upon termination of this action without further order of the Court, and are the persons to whom such confidential exhibits and other confidential matters may be returned by the Clerk of the Court, if they are not so retrieved. No confidential exhibits or other confidential matters shall be released in any other manner, except by order of the Court. Notwithstanding the foregoing, for material designated "Confidential – Outside Counsel Only – Source Code," the provisions of Section I.C are controlling to the extent those provisions differ from this Paragraph.

35. Protected Information shall not be copied or otherwise produced by a Receiving Party, except for transmission to qualified recipients, without the written permission of the Producing Party or by order of the Court. Nothing herein shall, however, restrict a qualified

recipient from making working copies, abstracts, digests, and analyses of “Confidential” and “Confidential – Outside Counsel Only” information for use in connection with this action. Such working copies, abstracts, digests, and analyses shall be deemed Protected Information under this Order. Further, nothing herein shall restrict a qualified recipient from converting or translating “Confidential” and “Confidential – Outside Counsel Only” information into machine-readable form for incorporation into a data retrieval system used in connection with this action, provided that access to that Protected Information, in whatever form stored or reproduced, shall be limited to people qualified to receive such Protected Information under this Order.

36. The Receiving Party acknowledges that Protected Information received under this Order may be subject to export controls under the laws of the United States and other applicable laws. The Receiving Party shall comply with such laws and agrees not to knowingly export, re-export, or transfer Protected Information of the Producing Party without first obtaining all required United States or any other applicable authorizations or licenses. Without limitation, this prohibition extends to Protected Information (including copies) in physical and electronic form. Notwithstanding this prohibition, and to the extent otherwise permitted by law, Protected Information, exclusive of material designated “Confidential – Outside Counsel Only – Source Code,” may be taken outside the territorial limits of the United States if it is reasonably necessary for a deposition taken in a foreign country. The restrictions contained within this Section may be amended through the consent of the Producing Party to the extent that such agreed-to procedures conform with applicable export control laws and regulations.

37. The Receiving Party agrees to maintain adequate controls to prevent nationals of countries listed in the EAR, Part 740 Supplement No. 1, Country Group D:1 or E from accessing the Producing Party’s Protected Information, subject to ECCN 5E001; or nationals outside the

United States and Canada from accessing such Protected Information, subject to ECCN 5E002 – without U.S. Government authorization. The Receiving Party further agrees to notify the Producing Party prior to granting a foreign national of countries listed in the groups D:1 or E access to the Source Code Computers, access to hard copies of Protected Information, or placement on a project requiring receipt or review of the Producing Party's Protected Information.

VII. NON-PARTY USE OF THIS PROTECTIVE ORDER

38. A non-Party producing information or material voluntarily or pursuant to a subpoena or a court order may designate such material or information as Protected Information pursuant to the terms of this Order. By doing so, the non-Party agrees to be bound by any applicable terms of this Order.

39. A non-Party's use of this Order to protect its Protected Information does not entitle that non-Party access to the Protected Information produced in this action.

VIII. NO WAIVER OF PRIVILEGE

40. Nothing in this Order shall require production of information that a Party or non-Party contends is protected from disclosure by the attorney-client privilege, the work product immunity, or other privilege, doctrine, right, or immunity. If information subject to a claim of attorney-client privilege, work product immunity, or other privilege, doctrine, right, or immunity ("Privileged Material") is nevertheless inadvertently or unintentionally produced, such production shall in no way prejudice or otherwise constitute a waiver or estoppel as to any such privilege, doctrine, right, or immunity. Upon becoming aware of the production of Privileged Material, the Producing Party or non-Party must promptly notify the recipient(s) of such inadvertent production in writing. Upon receipt of such notice, the recipient(s) shall gather and destroy all copies of the claimed Privileged Material and certify that it has done so to the

Producing Party within seven days of receipt of the notice. Notwithstanding this provision, outside counsel of record are not required to delete information that may reside on their respective firm's electronic back-up systems that are over-written in the normal course of business.

41. Within 14 days of the Producing Party's notice of inadvertent production, the Producing Party shall provide a privilege log identifying the inadvertently produced Privileged Material. The Receiving Party may move the Court for an order compelling production of such Privileged Material in accordance with the Federal Rules of Civil Procedure. Such a motion to compel shall be filed under seal and shall not assert as a ground for production the fact of the inadvertent production, nor shall the motion disclose or otherwise use the content of the inadvertently produced Privileged Material in any way beyond that which is reasonably necessary to identify the Privileged Material and its nature for the Court.

IX. INADVERTENT DISCLOSURE NOT AUTHORIZED BY ORDER

42. In the event of a disclosure of any Protected Information pursuant to this Order to any person not authorized to receive such material under this Order, the Party responsible for such disclosure – and each Party with knowledge of such disclosure – shall immediately notify outside counsel for the Producing Party whose Protected Information has been disclosed and provide to such counsel all known relevant information concerning the disclosure. The responsible or knowledgeable Party also shall promptly take all reasonable measures to retrieve the improperly disclosed Protected Information and to ensure that no further unauthorized disclosure or use thereof is made, including requesting an agreement from the recipients not to further disseminate or use the Protected Information in any form.

43. Compliance with the preceding Paragraph shall not prevent the Producing Party from seeking further relief from the Court.

44. Unauthorized or inadvertent disclosure does not waive or alter the protected status of disclosed Protected Information.

X. MISCELLANEOUS PROVISIONS

45. Any of the notice requirements herein may be waived, in whole or in part, but only in writing signed by an attorney for the Party against whom such waiver will be effective.

46. Any document served or filed that contains Protected Information (excluding exhibits or attachments thereto) shall bear a Certificate of Confidentiality (or otherwise be marked on its cover page) indicating which Party's or non-Party's Protected Information is contained therein. The Certificate shall be substantially similar to the following form: "This document contains confidential information subject to the protective order entered in this action. The confidential information contained herein is that of [identify each Party and/or non-Party as appropriate]." The absence of a Certificate of Confidentiality or marking of confidentiality on the cover page shall constitute a representation of the serving or filing Party that such document contains no Protected Information. In addition, if a document filed contains Protected Information of the filing Party, or a non-Party, the filing Party shall file a version of the document redacting the Protected Information of the filing Party or non-Party within five business days of service.

47. **Inadvertent failure to properly designate.** Inadvertent or unintentional production of documents or things containing Protected Information which are not designated as one or more of the three categories of Protected Information at the time of production shall not be deemed a waiver in whole or in part of a claim for confidential treatment. With respect to documents, the Producing Party shall notify the other Parties of the error in writing within 14 days of discovery of the error, and, within seven days of notifying the other Parties, provide replacement pages bearing the appropriate confidentiality designation. Upon receiving the

reproduced Protected Information, the Receiving Parties shall destroy all copies of the material that was not designated properly. In the event of any disclosure of Protected Information other than in a manner authorized by this Order, including any unintentional or inadvertent disclosure, counsel for the Party responsible for the disclosure shall immediately notify the Producing Party and all counsel of record in this action of all of the pertinent facts, and make every effort to further prevent unauthorized disclosure, including retrieving all copies of the Protected Information from the recipient(s) and securing the agreement of the recipient(s) not to further disseminate the Protected Information in any form. Compliance with the foregoing shall not prevent the Producing Party from seeking further relief from the Court.

48. Any person who reviewed Protected Information prior to a later designation of “Confidential – Outside Counsel Only” or “Confidential – Outside Counsel Only – Source Code” under the preceding Paragraph shall not be prohibited from engaging in the activities set forth in Section III (Prosecution Bar) on the basis of such Protected Information.

49. Within 60 days after the entry of a final non-appealable judgment or order, or the complete settlement of all claims asserted against all Parties in this action, each Party shall, at the option of the Producing Party, either return or destroy all Protected Information, including without limitation physical objects, documents, and electronic information or files containing Protected Information, and shall also destroy correspondence, memoranda, notes, and other work product materials, which contain or refer to any Protected Information. In the event that a Party is dismissed before the entry of a final non-appealable judgment or order, this same procedure shall apply to any Protected Information received from or produced to the dismissed Party after entry of a final non-appealable judgment or order, or the complete settlement of all claims asserted against all remaining Parties in this action. If a Producing Party opts to have all

Receiving Parties destroy Protected Information, each Receiving Party must certify its destruction to the Producing Party. Notwithstanding this provision, outside counsel of record are not required to delete information that may reside on their respective firm's electronic back-up systems that are over-written in the normal course of business. Notwithstanding the foregoing, outside counsel shall be entitled to maintain only for archival purposes copies of all correspondence, pleadings, motions, and trial briefs (including all supporting and opposing papers and exhibits thereto), written discovery requests and responses (and exhibits thereto), deposition transcripts (and exhibits thereto), trial transcripts, and exhibits offered or introduced into evidence at any hearing or trial, and their attorney work product which refers or is related to any "Confidential" and "Confidential – Outside Counsel Only" information.

50. If at any time documents containing Protected Information are subpoenaed by any court, arbitral, administrative, or legislative body, or are otherwise requested in discovery, the person or Party to whom the subpoena or other request is directed shall within five business days give written notice thereof to every Party who has produced such documents and to its counsel and shall provide each such Party with an opportunity to object to the production of such documents. If a Producing Party does not seek a protective order with respect to such documents within ten business days of the date written notice is given, the person or Party to whom the referenced subpoena is directed may produce such documents in response thereto, but shall take all reasonable measures to have such documents treated in accordance with terms of this Order.

51. Testifying experts shall not be subject to discovery of any draft of their reports in this action and such draft reports, notes, outlines, or any other writings leading up to an issued report(s) in this action are considered attorney work-product and are exempt from discovery. In addition, all communications between counsel for a Party and that Party's testifying expert, and

all materials generated by a testifying expert with respect to that person's work, are also considered attorney work-product and are exempt from discovery unless they relate to the expert's compensation or identify facts, data, or assumptions relied upon by the expert in forming any opinions in this action and such information is not already disclosed in the expert's report.

52. No Party shall be required to identify on their respective privilege log any document or communication dated on or after the filing of the lawsuit. The Parties shall exchange their respective privilege logs at a time to be agreed upon by the Parties following the production of documents.

53. This Order is entered without prejudice to the right of any Party to apply to the Court at any time for additional protection, or to relax or rescind the restrictions of this Order, when convenience or necessity requires. Furthermore, without application to the Court, the Parties may enter a written agreement releasing one or more Parties and any individual subject to this agreement from one or more requirements of this Order.

54. All disputes concerning Protected Information produced under this Order shall be resolved by the United States District Court for the District of Massachusetts. After termination of this action, the provisions of this Order shall continue to be binding, and the Court shall retain jurisdiction over the Parties and recipients of Protected Information for enforcement of the provisions of this Order, including after the termination of this action.

55. Nothing in this Order shall preclude or impede outside counsel of record's ability to communicate with or advise their client in connection with this action based on such counsel's review and evaluation of Protected Information, provided, however, that such communications or

advice shall not disclose or reveal the substance or content of any Protected Information other than as permitted under this Order.

56. Each of the Parties agrees to be bound by the terms of this Order as of the date counsel for such Party executes this Order, even if prior to entry of this Order by the Court. This Order shall be binding on the Parties, their attorneys, and their successors, executors, personal representatives, administrators, heirs, legal representatives, assigns, subsidiaries, divisions, employees, agents, and retained consultants, experts, and any persons or organizations over which they have direct control.

57. Protected Information must be stored and maintained by a Receiving Party in a secure manner that ensures that access is limited to the persons authorized under this Order.

58. The computation of any period of time prescribed or allowed by this Order shall be governed by the provisions for computing time set forth in Federal Rule of Civil Procedure 6.

59. Nothing in this Order shall be construed as an admission by any Party that any particular type of information is relevant or discoverable.

Respectfully submitted:

/s/ Avery R. Williams

Mike McKool (*admitted pro hac vice*)
TX Bar No. 13732100
mmckool@mckoolsmith.com
Christopher T. Bovenkamp (*admitted pro hac vice*)
TX Bar No. 24006877
cbovenkamp@mckoolsmith.com
Avery R. Williams (*admitted pro hac vice*)
TX Bar No. 24075282
awilliams@mckoolsmith.com
MCKOOL SMITH, P.C.
300 Crescent Court Suite 1500
Dallas, TX 75201
Telephone: (214) 978-4000
Fax: (214) 978-4044

John B. Campbell (*admitted pro hac vice*)
TX Bar No. 24036314
jcampbell@mckoolsmith.com
James. E. Quigley (*admitted pro hac vice*)
TX Bar No. 24075810
jqigley@mckoolsmith.com
MCKOOL SMITH, P.C.
300 W. 6th Street, Suite 1700
Austin, TX 78701
Telephone: (512) 692-8700
Fax: (512) 692-8744

David L. Evans (BBO #156695)
devans@murphyking.com
Steven M. Veenema (BBO #672097)
sveenema@murphyking.com
MURPHY & KING, P.C.
One Beacon Street, 21st Fl.
Boston, Massachusetts 02108-3107
Telephone: (617) 423-0400
Fax: (617) 423-0498

Counsel for Plaintiff Egenera, Inc.

IT IS SO ORDERED.

Date: 7-17-17

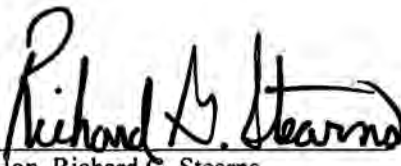
/s/ Peter C. Magic

John M. Desmarais (*admitted pro hac vice*)
jdesmarais@desmaraisllp.com
Paul A. Bondor (*admitted pro hac vice*)
pbondor@desmaraisllp.com
Jonas R. McDavit (*admitted pro hac vice*)
jmcavit@desmaraisllp.com
Tamir Packin (*admitted pro hac vice*)
tpackin@desmaraisllp.com
Peter C. Magic (*admitted pro hac vice*)
pmagic@desmaraisllp.com
Brian Leary (*admitted pro hac vice*)
bleary@desmaraisllp.com
Michael R. Rhodes (*admitted pro hac vice*)
mrhodes@desmaraisllp.com
DESMARAIS LLP
230 Park Avenue
New York, NY 10169
Telephone: (212) 351-3400
Facsimile: (212) 351-3401

/s/ John W. Moran

Kevin G. Kenneally (BBO # 550050)
Kevin.Kenneally@leclairryan.com
John W. Moran (BBO # 664914)
John.Moran@leclairryan.com
LECLAIRRYAN
One International Place, Suite 1110
Boston, Massachusetts 02110
Telephone: (617) 502-8220
Facsimile: (617) 502-8270

Counsel for Defendant Cisco Systems, Inc.


Hon. Richard G. Stearns
United States District Judge

ATTACHMENT A
TO THE PROTECTIVE ORDER REGARDING PROTECTED INFORMATION IN
***Egenera, Inc. v. Cisco Systems, Inc.*, No. 1:16-cv-11613-RGS (D. Mass.)**

1. My name is _____.
2. I reside at _____.
3. I am a citizen of the following country(ies): _____.
4. My present employer is _____.
5. My present employer's address is _____.
6. My present occupation or job description is _____.
7. I have read the Protective Order Regarding Protected Information dated _____, 2017, and have been engaged as _____ on behalf of _____ in the preparation and conduct of litigation styled *Egenera, Inc. v. Cisco Systems, Inc.*, No. 1:16-cv-11613-RGS (D. Mass.).

8. I am fully familiar with and agree to comply with and be bound by the provisions of said Order. I understand that I am to retain all copies of any documents designated as "Confidential," "Confidential – Outside Counsel Only," or "Confidential – Outside Counsel Only – Source Code," or any similar designation, in a secure manner, and that all copies are to remain in my personal custody until I have completed my assigned duties, whereupon the copies and any writings prepared by me containing any information designated "Confidential," "Confidential – Outside Counsel Only," or "Confidential – Outside Counsel Only – Source Code," or any similar designation, are to be returned to counsel who provided me with such material.

9. I will not divulge to persons other than those specifically authorized by said Order, and will not copy or use except solely for the purpose of this action, any information

obtained pursuant to said Order, except as provided in said Order. I also agree to notify any stenographic or clerical personnel who are required to assist me of the terms of said Order.

10. I state under penalty of perjury under the laws of the United States of America that the foregoing is true and correct.

By: _____

Executed on _____, 20____.

UNITED STATES DISTRICT COURT
DISTRICT OF MASSACHUSETTS

CIVIL ACTION NO. 16-11613-RGS

Egenera, Inc.

v.

Cisco Systems, Inc.

MEMORANDUM AND ORDER ON
CROSS MOTIONS FOR SUMMARY JUDGMENT AND
TO EXCLUDE EXPERT TESTIMONY

June 23, 2021

STEARNS, D.J.

Plaintiff Egenera, Inc., accuses defendant Cisco Systems, Inc., of infringing United States Patent No. 7,231,430 (the '430 patent). The case having returned to this court from the Court of Appeals for the Federal Circuit, the parties now cross move for a second round of summary judgment. Each side also seeks to exclude the testimony of their competing expert witnesses.

PROCEDURAL HISTORY

Egenera filed its Complaint for patent infringement in August of 2016.¹ In April of 2017, Cisco petitioned the PTAB to institute an IPR of the '430

¹ In its initial Complaint, Egenera also asserted infringement of U.S. Patents Nos. 6,971,044 (the '044 patent) and 7,178,059 (the '059 patent). On

patent. While the petition was pending, Egenera withdrew Peter Schulter as a named co-inventor of the patent. *See Egenera, Inc. v. Cisco Sys., Inc.*, 379 F. Supp. 3d 110, 113-114 ¶¶ 10-18 (D. Mass. 2019) (Inventorship Rulings). In February of 2018, the court construed the disputed claim terms and concluded, *inter alia*, that the “logic to modify” term was means-plus-function embodying a tripartite structure of “virtual LAN server 335, virtual LAN proxy 340, and physical LAN driver 345.” *See Egenera, Inc. v. Cisco Sys., Inc.*, 2018 WL 717342, at *4-7 (D. Mass. Feb. 5, 2018) (CC Order).²

Cisco’s motion to dismiss, the court found the ’059 patent to be directed to patent-ineligible subject matter. *Egenera, Inc. v. Cisco Sys., Inc.*, 234 F. Supp. 3d 331, 345-346 (D. Mass. 2017) (MTD Opinion). Egenera dismissed the ’044 patent without prejudice after the Patent Trial and Appeal Board (PTAB) instituted *inter partes* review (IPR) on all claims. *See* Dkt ## 77 at 11-12; 78, 80, and 81.

² The full claim term is “logic to modify said received messages to transmit said modified messages to the external communication network and to the external storage network.” The court rejected Egenera’s argument that “logic” denotes “software, firmware, circuitry, or some combination thereof,” and instead determined that, because the term did not recite sufficient structure, it would be construed as means-plus-function. CC Order, at *4-6. The court concluded that “[t]he structure for modifying and transmitting messages to the external communications network is [] ‘virtual LAN server 335, virtual LAN proxy 340, and physical LAN driver 345’ and equivalents,” and “the structure for modifying and transmitting messages to the external storage network is ‘storage configuration logic 605’ and equivalents.” CC Order, at *7.

After the close of discovery, Cisco moved, *inter alia*, to invalidate the patent on grounds of the allegedly improper withdrawal of Schuler as a named inventor. In Cisco's view, Schuler had "contribute[d] to the conception of the claimed invention" as the originator of the tripartite structure. *Eli Lilly & Co. v. Aradigm Corp.*, 376 F.3d 1352, 1359 (Fed. Cir. 2004). The court agreed with Cisco that judicial estoppel barred Egenera from a tactical restoration of Schuler as an inventor, *see Egenera, Inc. v. Cisco Sys., Inc.*, 348 F. Supp. 3d 99, 101-102 (D. Mass. 2018), but concluded that sufficient disputes of fact remained to preclude an award of summary judgment, *see id.* at 108. Following a three-day bench trial, the court made detailed findings determining that Schuler had conceived the tripartite structure and was therefore a true inventor of the '430 patent. Thus, his elimination as an inventor invalidated the patent. Inventorship Rulings at 128-129 ¶¶ 83-84.

Egenera appealed. The Court of Appeals for the Federal Circuit held that Egenera's dropping of Schuler from the roster of inventors was a correctable error, and that judicial estoppel did not apply in the circumstances of the case. *See Egenera, Inc. v. Cisco Sys., Inc.*, 972 F.3d 1367, 1376-1381 (Fed. Cir. 2020) (CAFC Opinion). The Court, on the other

hand, affirmed this court's means-plus-function construction of the "logic to modify" term. *See id.* at 1372-1376.

Now back before this court on remand, Egenera moves for partial summary judgment of no "unclean hands" and no anticipation, and to strike the reasonable royalty opinions of Dr. Stephen Becker.³ Cisco counter-moves for summary judgment of unclean hands; noninfringement; non-entitlement to injunctive relief and pre-suit damages for indirect or willful infringement; and to strike the infringement opinions of Dr. Mark Jones and the reasonable royalty opinions of Dr. Ryan Sullivan.

CROSS MOTIONS FOR JUDGMENT AS TO UNCLEAN HANDS

Relying on testimony elicited at the inventorship trial, Cisco accuses Egenera of unclean hands. "[A] determination of unclean hands may be reached when 'misconduct' of a party seeking relief 'has immediate and necessary relation to the equity that he seeks in respect of the matter in litigation,' *i.e.*, 'for such violations of conscience as in some measure affect the equitable relations between the parties in respect of something brought before the court.'" *Gilead Scis., Inc. v. Merck & Co.*, 888 F.3d 1231, 1239 (Fed. Cir. 2018), quoting *Keystone Driller Co. v. Gen. Excavator Co.*, 290

³ In December of 2020, in light of the Federal Circuit's mandate, the court allowed Egenera's motion to correct the inventorship to reinstate Schuler. *See* Dkt # 318.

U.S. 240, 245 (1933). In Cisco's view, Egenera committed egregious litigation misconduct when four inventors of the '430 patent, enlisted by Egenera as paid consultants and represented by Egenera's counsel, testified falsely at the inventorship trial that Peter Schulter was not an inventor, contradicting at times contemporaneous documents that they themselves had authored. This testimony "ha[d] immediate and necessary relation" to the litigation because Egenera was desperate to preserve the validity of the '430 patent and its claims against Cisco.⁴

As Cisco accurately points out, the court did not credit the inventors' testimony minimizing Schulter's role in the creation of the invention and characterized it as "post-hoc protestations" and an exercise in "historical revisionism." Inventorship Rulings at 129, ¶ 83(g). Nevertheless, the court is unable to find that Egenera's sketchy posturing of the '430 patent's "Eureka moment" rose to the level of egregious misconduct that would warrant the drastic remedy of dismissal. As the Federal Circuit noted, Egenera's account of the inventorship was staked out at a time when neither party had advocated for a means-plus-function understanding of the "logic

⁴ Cisco also notes that, by excluding Schulter, the last of the inventors to be hired by Egenera as a member of the '430 patent team, Egenera could claim an earlier priority date to skirt a problematic prior art reference.

to modify” term and was thus “consistent with its preferred claim construction.” CAFC Opinion at 1377. Thereafter, Egenera was locked into its position owing in part to, as it turned out, this court’s erroneous application of judicial estoppel.⁵ As was the case here, inventorship “sometimes [] is complicated.” *Id.* at 1376. “Ultimately, inventorship is a legal conclusion premised on underlying factual findings, and one that depends on claim construction.” *Id.* The interplay of claim construction and inventorship in this case was settled only after “a three-day trial and [an] appeal.” *Id.* at 1378. Against this backdrop, while the court by no means endorses Egenera’s less than level downplaying of Schuler’s contribution to the ’430 patent, the court also cannot, in light of the Federal Circuit’s ruling, go so far as to conclude that the dictates of equity require dismissal. Accordingly, Cisco’s motion for summary judgment of unclean hands will be denied, and Egenera’s motion for summary judgment of unsoiled hands will be allowed.

⁵ Prior to the court’s judicial estoppel ruling, Egenera had advocated that correction and not invalidation was the appropriate remedy for misjoinder of inventors. *See* Dkt # 136 at 13.

CISCO'S MOTION FOR JUDGMENT OF NONINFRINGEMENT

Cisco contends that, in light of the evidentiary record and the court's claim construction, Egenera cannot plausibly make out a case of infringement. "To support a summary judgment of noninfringement it must be shown that, on the correct claim construction, no reasonable jury could have found infringement on the undisputed facts or when all reasonable factual inferences are drawn in favor of the patentee." *Netword, LLC v. Centraal Corp.*, 242 F.3d 1347, 1353 (Fed. Cir. 2001). Infringement comes in two flavors. "To establish literal infringement, all of the elements of the claim, as correctly construed, must be present in the accused system." *Id.* "For infringement by equivalency, all of the elements of the claimed invention or an equivalent thereof must be present in the accused system." *Id.* at 1354.

The '430 patent is directed to solving problems in manually configuring, deploying, and maintaining enterprise and application servers, see '430 patent, col. 1, ll. 21-58, and discloses "a processing platform from which virtual systems may be deployed through configuration commands," *id.*, col. 2, ll. 45-47.

The platform provides a large pool of processors from which a subset may be selected and configured through software commands to form a virtualized network of computers ("processing area network" or "processor clusters") that may be

deployed to serve a given set of applications or customer. The virtualized processing area network (PAN) may then be used to execute customer specific applications, such as web-based server applications. The virtualization may include virtualization of local area networks (LANs) or the virtualization of I/O storage. By providing such a platform, processing resources may be deployed rapidly and easily through software via configuration commands, e.g., from an administrator, rather than through physically providing servers, cabling network and storage connections, providing power to each server and so forth.

Id., col. 2, ll. 47-62.⁶

Egenera asserts claims 1, 3-5, and 7-8 of the '430 patent. Claim 1 is representative.

1. A platform for automatically deploying at least one virtual processing area network, in response to software commands, said platform comprising:

a plurality of computer processors connected to an internal communication network;

at least one control node in communication with an external communication network and in communication with an external storage network having an external storage address space, wherein the at least one control node is connected to the internal communication network and thereby in communication with the plurality of computer processors, said at least one control node including logic to receive messages from the plurality of computer processors, wherein said received messages are addressed to the external communication network and to the external storage network and said at least one control node including logic to modify said received messages to transmit said

⁶ Additional descriptions of the claimed invention of the '430 patent may be found in the court's Memorandum and Order on Cisco's motion to dismiss. See MTD Opinion at 334-336.

modified messages to the external communication network and to the external storage network;

configuration logic for receiving and responding to said software commands, said software commands specifying (i) a number of processors for a virtual processing area network (ii) a virtual local area network topology defining interconnectivity and switching functionality among the specified processors of the virtual processing area network, and (iii) a virtual storage space for the virtual processing area network, said configuration logic including logic to select, under programmatic control, a corresponding set of computer processors from the plurality of computer processors, to program said corresponding set of computer processors and the internal communication network to establish the specified virtual local area network topology, and to program the at least one control node to define a virtual storage space for the virtual processing area network, said virtual storage space having a defined correspondence to a subset of the external storage address space of the external storage network; and

wherein the plurality of computer processors and the at least one control node include network emulation logic to emulate Ethernet functionality over the internal communication network.

As pertains to this motion, in its claim construction the court rejected Egenera's proposal to equate "computer processor/processor" to a "processing node," and instead construed the term to encompass a "CPU." CC Order, at *2-4.

Cisco's accused Unified Computing System (UCS) is a "scalable compute platform." Egenera Ex. 1 (Dkt # 172-1) at 41. Components of UCS include the UCS Manager, Fabric Interconnects, Fabric Extenders and I/O

Modules, B-Series Blades and C-Series Rack Servers, and I/O adapters. Each of the asserted claims recites the limitation “software commands specifying . . . a number of processors for a virtual processing area network.” Egenera identifies the configConfMos software command as meeting this limitation. In Cisco’s view, because configConfMos contains no field identifying a number (of CPUs or anything else), it does not satisfy the claim limitation. Further, Cisco notes that, as configConfMos associates a service profile with a blade server – a “processing node” in the jargon of the patent – Egenera’s infringement theory ignores the court’s claim construction of “processor” as a “CPU.”

Egenera does not dispute that configConfMos does not identify an explicit numerical CPU parameter, but maintains that the command nevertheless satisfies the claim limitation. Egenera contends that because the number of CPUs in each Cisco blade server is known – it is revealed by the number following the series-identifier B- or C- in the server’s model number⁷ – by associating a particular server, configConfMos specifies a known number of CPUs for the UCS. Egenera also notes that, when a server

⁷ Egenera explains, for example, that the Cisco server with model number C460 has 4 CPU sockets, and that deploying a server with fewer CPUs than sockets could cause serious problems.

is added to a UCS, the UCS discovers the properties of the server, including the number of CPUs on the server. The UCS Service Profile of a server, further, displays the number of CPUs on the server.

The court cannot conclude as a matter of law that configConfMos does not meet the asserted limitation. The parties did not seek a construction for “specifying . . . a number of processors.” While the claim language can be read, as Cisco suggests, to require a specific numerical quantity, it can also be understood as identifying some number of processors as a group or selecting a group of specific processors.⁸ Cisco does not point to any support in the patent that would compel the specific value interpretation. The language in the specification, explaining that “[e]ach PAN, through software commands, is configured to have a corresponding *subset of processors*,” ’430 patent, col. 3, ll. 55-56 (emphasis added), is also consistent with the proposed less restrictive reading of the claim limitation permitting a factfinder to conclude that configConfMos specifies a number of CPUs for inclusion in the UCS, albeit indirectly, by associating a server with a known number of CPUs. *See Mentor Graphics Corp. v. EVE-USA, Inc.*, 851 F.3d 1275, 1282 (Fed. Cir. 2017) (where the parties did not seek to construe a claim limitation to

⁸ To take a mundane example, a request to “specify[] a number of donuts” could be satisfied with a response of “twelve,” “that box,” or “chocolate dipped, Boston cream, and apple fritter.”

indicate an RTL statement, the jury's infringement finding was supported by substantial evidence that the accused method generated a test file from which an RTL statement could be ascertained).

Cisco also seeks judgment of noninfringement of claims 1 and 5 on another ground. Claim 1 recites "the plurality of computer processors . . . include network emulation logic to emulate Ethernet functionality over the internal communication network." Claim 5 recites "the plurality of computer processors . . . emulate Ethernet functionality over the internal communication network." Cisco does not dispute that UCS emulates Ethernet functionality (at least for purposes of this motion) but contends that because Ethernet emulation functionality resides with virtual network interface cards (NIC) and interfaces — stand-alone components separate and apart from the CPUs — the limitations are not met.

Egenera points out that in each of claims 1 and 5, the Ethernet emulation functionality is attributed to "the plurality of computer processors *and at least one control node*." (emphasis added). It therefore follows that the emulation functionality is not required to reside uniquely on the CPUs. Egenera contends that UCS Server CPUs satisfy the claim limitation because they "communicate on and use virtual interfaces between themselves and

UCS Fabric Interconnects over the UCS internal communication network.” Egenera Opp’n (Dkt # 171) at 17.

While Egenera is correct that Ethernet emulation functionality need not reside on the CPUs alone, the claims nonetheless require the CPUs to include some logic to emulate Ethernet functionality or to emulate Ethernet functionality in some respect. The extent of the CPU’s role, as Egenera explains, is its “knowledge and use of the virtual MAC address (and other related information) over the virtual interface.” *Id.* at 19. However, knowledge and use of a communications network is not emulation of the functionality of that network – a person dialing and making a telephone call to another’s phone number merely uses a telephone network and does not emulate any functionality of that network. Egenera identifies no evidence that the CPUs in the UCS provide any aspect of the functionality of an Ethernet network. The court will accordingly allow summary judgment of noninfringement on claims 1 and 5.

CISCO’S MOTION TO EXCLUDE THE INFRINGEMENT
OPINIONS OF DR. JONES

Cicso seeks to exclude the infringement opinions of Egenera’s expert witness, Dr. Mark Jones, on the grounds that he disregarded the court’s construction of the term “computer processor/processor” as a CPU and improperly equated it to a processing node. While the court agrees with

Cisco that an expert witness must apply the court's claim construction in his or her infringement and invalidity analyses, *see Exergen Corp. v. Wal-Mart Stores, Inc.*, 575 F.3d 1312, 1321 (Fed. Cir. 2009), the court disagrees that Dr. Jones contravened this rule. As explained earlier, Dr. Jones's theory of how the accused UCS satisfies the "specifying a number of processors" limitation is at least a plausible reading of the claim language.

In a footnote, Cisco also challenges Dr. Jones's analysis of the limitation "defining interconnectivity and switching functionality among the specified processors," contending that the virtual NICs, rather than the CPUs, defined the network topology of the UCS. However, as Cisco acknowledged during claim construction in advocating for the CPU construction of "computer processor," nothing in the patent requires a direct connection between computer processors.

Cisco's final example of an alleged breach by Dr. Jones of the claim construction is a diagram presented in paragraph 72 of his report that was also included in Egenera's claim construction presentation. In this diagram, a group of processor nodes are block-colored and labeled as "computer processors." This diagram can be understood, as Egenera advocates, as indicating the location of "computer processors" on the processing nodes. While the court agrees, to avoid any potential of confusion on the part of

jurors, it will direct Egenera to make the simple adjustment of corresponding the label of “computer processors” with CPUs (106j and 106l). Subject to this prophylactic, Cisco’s motion will be denied.

EGENERA’S MOTION FOR JUDGMENT OF NO ANTICIPATION

Egenera seeks judgment of no anticipation as a matter of law. To establish anticipation invalidity, “the four corners of a single[] prior art document [must] describe every element of the claimed invention, either expressly or inherently.” *TriMed, Inc. v. Stryker Corp.*, 608 F.3d 1333, 1343 (Fed. Cir. 2010). A claim of patent invalidity must be proven by clear and convincing evidence. *Microsoft Corp. v. i4i Ltd. P’ship*, 564 U.S. 91, 95 (2011).

Egenera contends that each of Cisco’s prior art references is missing at least one claim element – “a plurality of computer processors and at least one control node connected to an internal communication network.” ’430 patent claims 5, 7, and 8.⁹ Egenera notes that the PTAB, using a broader claim construction standard and a lower burden of proof, declined to institute on Cisco’s petition for IPR of the ’430 patent because Cisco did not

⁹ Claims 1, 3, and 4, the remaining independent claims, similarly require that “the at least one control node is connected to the internal communication network and thereby in communication with the plurality of computer processors.”

sufficiently establish that the asserted references taught a control node, or a control node connected to an internal network. Egenera asserts that Cisco's anticipation contentions in this case suffer from the same deficiency.

Cisco responds by pointing to the anticipation analysis of its expert witness, Dr. Kevin Jeffay, including the element-by-element charts for each asserted prior art reference or system. For example, Dr. Jeffay explains that the Cisco Catalyst System discloses a control node connected to an internal communications network and a plurality of computer processors because it “connects a plurality of computers to one or more Catalyst switches and/or routers.” Cisco Ex. 64 (Dkt # 175-1) at 48.¹⁰ Cisco also distinguishes the PTAB's denial of institution of the IPR because the IPR concerned obviousness arguments rather than anticipation, and because it now asserts art that was not before the PTAB.¹¹

¹⁰ Cisco adds the further clarification that, in the Catalyst System, a switch is a control node connected to computer processor(s) through a communication network of wiring. See Cisco Opp'n (Dkt # 325) at 19.

¹¹ Cisco identifies thirteen pieces of alleged anticipatory prior art, see Cisco Opp'n at 19, while the denial of IPR institution was based on only three prior art patents, see Egenera Ex. 12 (Dkt # 312-12) at 6. In any case, a decision by the PTAB to deny institution of IPR does not estop a party from raising the same arguments before the district court. See *Shaw Indus. Grp., Inc. v. Automated Creel Sys., Inc.*, 817 F.3d 1293, 1300 (Fed. Cir. 2016).

In reply, Egenera faults Cisco for “conflat[ing]” “two claimed components [] into a single component for purposes of a prior art analysis.” Egenera Reply (Dkt # 328) at 5, citing *Becton, Dickinson & Co. v. Tyco Healthcare Grp., LP*, 616 F.3d 1249, 1254 (Fed. Cir. 2010) (“Where a claim lists elements separately, ‘the clear implication of the claim language’ is that those elements are ‘distinct component[s]’ of the patented invention.”) (citation omitted). Courts have not, however, followed *Becton* literally where, as here, the asserted patent concerns a computer implemented system. In *Intellectual Ventures I LLC v. Symantec Corp.*, 2016 WL 948879 (D. Del. Mar. 10, 2016), *aff’d*, 725 F. App’x 976 (Fed. Cir. 2018), the court rejected an argument based on *Becton* that the claim element “data transfer unit” (DTU) is necessarily physically separate and distinct from the claim element “network server” in a system directed to the remote mirroring of data. *See* 2016 WL 948879, at *3. “*Becton* involved physical components, whereas the DTU in the present invention undisputedly involves both hardware and software. Here, the claims involve digital, rather than physical separation.” *Id.*

Moreover, the ’430 patent does not mandate physical separation of the “control node” from other components. The only claimed physical requirement of the “control node” is that it be “connected to the internal

communication network and thereby in communication with the plurality of computer processors.” ’430 patent claim 1; *see also NTP, Inc. v. Research in Motion, Ltd.*, 418 F.3d 1282, 1310 (Fed. Cir. 2005) (“A ‘connection’ can occur between these two devices regardless of whether they are housed separately or together.”). The remaining requirements are functional: “said at least one control node including logic to receive messages from the plurality of computer processors,” and “said at least one control node including logic to modify said received messages to transmit said modified messages to the external communication network and to the external storage network.” ’430 patent claim 1. Accordingly, the court cannot conclude that Cisco’s anticipation contentions – to the extent that they map multiple claim elements to the same physical component – are deficient as a matter of law. Egenera’s motion for judgment of no anticipation will be denied.¹²

¹² In a single paragraph in its reply, Egenera argues that, in the context of the Catalyst System, the wiring connecting processors to a switch cannot constitute a programmable network as required by the patent. *See* ’430 patent claim 1 (“said configuration logic including logic . . . to program said corresponding set of computer processors and the internal communication network to establish the specified virtual local area network topology”). This contention has not been sufficiently briefed, nor is it clear that it is applicable to all of Cisco’s asserted anticipatory prior art. The court will therefore not consider it further.

CISCO'S MOTION FOR NO INJUNCTIVE RELIEF

Cisco asserts that Egenera cannot as a matter of law establish entitlement to injunctive relief, should Egenera prove infringement. “The[] familiar principles [of equity] apply with equal force to disputes arising under the Patent Act.” *eBay Inc. v. MercExchange, L.L.C.*, 547 U.S. 388, 391 (2006). To obtain injunctive relief,

[a] plaintiff must demonstrate: (1) that it has suffered an irreparable injury; (2) that remedies available at law, such as monetary damages, are inadequate to compensate for that injury; (3) that, considering the balance of hardships between the plaintiff and defendant, a remedy in equity is warranted; and (4) that the public interest would not be disserved by a permanent injunction.

Id.

In Cisco's view, Egenera's seven-year delay in initiating this lawsuit in 2016, after learning of UCS in 2009, undermines its claim of irreparable harm. Egenera also ceased selling its patent-embodiment BladeFrame systems in 2008. As Cisco sees it, because Egenera no longer competes in the server market, it cannot suffer any future harm, at least of an irreparable nature, from Cisco's sales of UCS. Compounding the issue, Egenera has allowed other players in the server market to sell rebranded versions of its products in exchange for pecuniary compensation, and has made a similar licensing offer to Cisco in the past. Egenera's willingness to license its

technology, Cisco fairly argues, reflects the adequacy of money damages. Cisco also points out that Egenera did not seek a preliminary injunction, does not seek lost profits in this case, and has already determined a reasonable royalty in the neighborhood of \$1,000 per unit of UCS. Finally, Cisco maintains that the balance of hardships favors it as an active participant in the market, and that the public has a greater interest in accessing its innovative products, especially given the fact that Egenera is unable to offer customers anything equivalent.

In response, Egenera asserts that it had only come to a firm conviction that Cisco had infringed its patented technology on the eve of filing suit, and that, further, it would be unfair to overemphasize any pre-suit delay in view of the “daunting task” faced by a smaller company like Egenera in enforcing its intellectual property rights against an industry giant like Cisco. Egenera Opp’n (Dkt # 180) at 6. Egenera also disputes Cisco’s characterization of its lack of market participation. Although it no longer markets servers, Egenera avers that it actively sells its PAN Manager software in combination with hardware from multiple manufacturing partners. PAN Manager, in Egenera’s view, serves the same function as the software on Cisco’s UCS. A purchase of UCS therefore displaces the purchase of a product that incorporates PAN Manager, and in that way, Cisco can be said to directly

compete with Egenera.¹³ See *Presidio Components, Inc. v. Am. Tech. Ceramics Corp.*, 702 F.3d 1351, 1363 (Fed. Cir. 2012) (“Even without practicing the claimed invention, the patentee can suffer irreparable injury. Direct competition in the same market is certainly one factor suggesting strongly the potential for irreparable harm without enforcement of the right to exclude.”).

Egenera also offers a different calculus of the relative hardships and public interest. Egenera fears that Cisco’s continual dominance in the server virtualization market (a position built on the alleged infringement, as Egenera sees it) would obliterate Egenera’s software business altogether. Egenera also notes that an injunction would not impact existing Cisco UCS users, and views the inability to purchase new Cisco UCS systems as only a minor irritant when weighed against the stronger public interest in protecting and promoting intellectual property and innovation.

Absent an infringement determination and in light of, *inter alia*, factual issues surrounding Egenera’s market participation, the court agrees with Egenera that it is premature to assess the availability of injunctive relief

¹³ Egenera suggests that an injunction can be narrowly tailored to enjoin only Cisco’s distribution of the offending software, but not the hardware itself (or with other software).

at this point in the litigation. Accordingly, the court will deny Cisco's motion subject to renewal post-trial on a perfected evidentiary record.

CISCO'S MOTION FOR JUDGMENT OF NO PRE-SUIT DAMAGES,
INDIRECT INFRINGEMENT, OR WILLFULNESS

Pre-Suit Damages

Cisco contends that Egenera is not entitled to pre-suit damages for any alleged infringement because Egenera did not mark its patent-embodiment products in accordance with 35 U.S.C. § 287(a). "Pursuant to 35 U.S.C. § 287(a), a patentee who makes or sells a patented article must mark his articles or notify infringers of his patent in order to recover damages." *Arctic Cat Inc. v. Bombardier Recreational Prods. Inc.*, 876 F.3d 1350, 1365 (Fed. Cir. 2017). Constructive notice through marking may be effectuated

either by fixing [on a patented article] the word "patent" or the abbreviation "pat.", together with the number of the patent, or by fixing thereon the word "patent" or the abbreviation "pat." together with an address of a posting on the Internet, accessible to the public without charge for accessing the address, that associates the patented article with the number of the patent, or when, from the character of the article, this can not be done, by fixing to it, or to the package wherein one or more of them is contained, a label containing a like notice.

35 U.S.C. § 287(a). Compliance with the marking statute is a question of fact with the burden of proof assigned to the patentee. *Arctic Cat*, 876 F.3d at 1366.

Egenera does not assert that it gave actual notice of the alleged infringement to Cisco prior to filing suit. At issue is whether Egenera provided sufficient constructive notice to open the door for an award of pre-suit damages. Cisco notes, and Egenera does not dispute, that Egenera's BladeFrame systems were not marked with the '430 patent number. Egenera instead contends that the constructive notice period began on October 1, 2013, after it had stopped selling physical servers. Egenera explains that, as of that date, it began virtually marking its Pan Manager software by including the following language in four user reference manuals¹⁴ provided with Pan Manager:

This product is protected by U.S. and international copyright and intellectual property laws. Egenera products are covered by one or more patents listed at <http://www.egenera.com/patents>.

The website, in turn, at substantially¹⁵ all times since August of 2012, listed Egenera's inventory of patents, including the '430 patent. *See Maxwell v. J. Baker, Inc.*, 86 F.3d 1098, 1111 (Fed. Cir. 1996) (“[O]nce marking has begun,

¹⁴ These include the Configuration and Installation Guide, the API Primer, the Command Reference, and the Administrator's Guide.

¹⁵ Egenera acknowledges that the website may have become unavailable between May and August of 2016 as the result of a website refurbishing.

it must be substantially consistent and continuous in order for the party to avail itself of the constructive notice provisions of the statute.”).

Egenera maintains that this virtual notice complied with the marking statute. As a practical matter, because Pan Manager was distributed as a downloadable image, there was no physical product or packaging on which a patent notice could have been inscribed. The reference manuals containing the notice were distributed (in media kits) with the downloadable software image, and users routinely consulted them during the installation and use of Pan Manager.

In the court’s view, Egnera’s argument misses the mark. Section 287(a) permits marking by label or on a package in lieu of “fixing” a notice on the patented article when “from the character of the article, [physical marking] can not be done.” Here, Egenera does not contend that Pan Manager by itself practices the ’430 patent. Rather, it is the combination of third-party hardware and the Pan Manager software that is asserted to embody the ’430 patent. Egenera does not dispute that a patent notice could have been physically placed on the third-party hardware. Egenera insists that it would have been improper for it to mark third-party hardware as the hardware is often sold without Pan Manager. Egenera, however, does not explain why third-party hardware installed with Pan Manager could not have

been appropriately marked by the hardware manufacturer or the distributor.¹⁶

A licensee, including an implied licensee, “who makes or sells a patented article does so ‘for or under’ the patentee, thereby limiting the patentee’s damage recovery when the patented article is not marked.” *Amsted Indus. Inc. v. Buckeye Steel Castings Co.*, 24 F.3d 178, 185 (Fed. Cir. 1994). In *Amsted*, the patentee sold a component of a patented device to customers “with the expectation that they would use that element to make and sell the patented invention,” rather than under an express license. *Id.* at 184. The Federal Circuit concluded that the patentee “could have sold its [component] with a requirement that its purchaser-licensees mark the patented products ‘licensed under U.S. X,XXX,XXX.’,” and that without such public notice of the patented article, the patentee could not recover pre-suit damages. *Id.* at 185. Here, Egenera implicitly authorized third-party manufacturers to sell hardware with Pan Manager installed, and the absence

¹⁶ Cisco notes that, in the case of at least one hardware manufacturer, a separate entity was hired to install Pan Manager on the hardware and distribute the (unmarked) assemblage to end-users. Cisco posits that the installer/distributor could easily have marked the finished product for Egenera’s benefit.

of marking on the patented combination precludes Egenera's pre-suit recovery.¹⁷

Even if, for argument's sake, off-product marking would suffice in this case, the substance of Egenera's constructive notice is nonetheless defective. Virtual marking, like physical marking, must provide public notice of *the patented article*. See *Nike, Inc. v. Wal-Mart Stores, Inc.*, 138 F.3d 1437, 1443 (Fed. Cir. 1998) ("The marking statute serves three related purposes: 1) helping to avoid innocent infringement, 2) encouraging patentees to give notice to the public that the article is patented, and 3) aiding the public to identify whether an article is patented.") (internal citations omitted). In *McKesson Automation, Inc. v. Swisslog Italia S.P.A.*, 712 F. Supp. 2d 283 (D. Del. 2010), a list of patents appeared on the login-screen of the software (Connect-Rx) that controlled the patented hardware system (Robot-Rx). *Id.* at 296. The court rejected the patentee's assertion that this passed muster. "[A] user has no way of knowing which patents listed on the log-in screen cover which of the multiple products controlled by the Connect-Rx software,

¹⁷ There is an exception that applies in the following circumstance: "[w]hen the failure to mark is caused by someone other than the patentee, the court may consider whether the patentee made reasonable efforts to ensure compliance with the marking requirements." *Maxwell*, 86 F.3d at 1111-1112. Egenera has not claimed any efforts to ensure the marking of the combination of third-party hardware and the Pan Manager software.

or whether the patents cover the Connect-Rx software itself.” *Id.* at 297. Because there was no clear association of the displayed patent numbers with any given patented article, “[t]he court conclude[d] that the marking displayed by the Connect-Rx software does not sufficiently apprise the public that the Robot-Rx is covered by the patents-in-suit.” *Id.* Here, a user would similarly not divine from the generic notice in a reference manual directed to “Egenera’s products” that the marriage of third-party hardware with Egenera’s Pan Manager comprised the patented article.

Virtual marking must also “provide such notice in a manner commensurate with the notice provided by physical marking,” *Mfg. Res. Int’l, Inc. v. Civiq Smartscapes, LLC*, 397 F. Supp. 3d 560, 577 (D. Del. 2019), that is, notice sufficient to “associate[] the patented article with the number of the patent,” *id.*, quoting 35 U.S.C. § 287(a). In *Manufacturing Resources*, the court found wanting a patentee’s marking website that listed the category of products covered by each patent but not the specific patents associated with each covered product. *Id.* at 577-578. “Mere direction to a general website listing patents for all the patentee’s products does not create this same association.” *Id.* at 577. Nor does “[s]imply listing all patents that could possibly apply to a product or all patents owned by the patentee on the

patentee's marking website[.] It merely creates a research project for the public." *Id.*

Egenera attempts to distinguish *Manufacturing Resources* by pointing to what it perceives as differences between the two websites at issue.¹⁸ Egenera notes that its website listed only 14 patents, all of which pertain to the server virtualization technology practiced by three of its four products (with the fourth product being unrelated), rather than the approximately 100 patents covering 46 products in *Manufacturing Resources*. The court does not find these distinctions any more meaningful than debating the number of possible states of a Rubik's cube. Egenera's webpage displays only a table of patent numbers and titles, and does not include the product information that it now seeks to rely on.¹⁹ Further, that a smaller number of patents entails a less time-consuming research project does not alter the fact that the

¹⁸ Egenera also cites to *National Prods., Inc v. Arkon Res., Inc*, 2019 WL 1034321 (C.D. Cal. Jan. 9, 2019) as authority supporting its position. In *National Products*, the court denied summary judgment of non-compliance with the marking statute where the patentee's website listed over 100 patents and did not identify the specific products associated with the asserted patents. In so holding, the court declined to engage in a substantive analysis based on a "limited record" for what appeared to be a question of first impression. *Id.*, at *16. That rationale is not applicable here.

¹⁹ Egenera does not assert that the three relevant products each practice all 14 patents.

webpage does not provide the statutorily required association between a patented product and the applicable patents. Finally, as the court in *Manufacturing Resources* noted, “permitting such a practice would likely create issues under the false marking statute if association could be inferred solely from marking the product with the website address.” 397 F. Supp. 3d at 577. Nothing on Egenera’s webpage informs a visitor which of its products practice the listed patents (or a subset thereof), and which do not. In sum, Egenera’s virtual marking does not sufficiently apprise the public that the combination of third-party hardware and Pan Manager is covered by the ’430 patent. *See Nike*, 138 F.3d at 1446 (The focus of the marking inquiry is “whether the patentee’s actions were sufficient, in the circumstances, to provide notice *in rem*.”).

Post-Suit²⁰ Indirect and Willful Infringement

Cisco asserts that Egenera cannot prove liability for post-suit indirect and willful infringement because of Cisco’s reasonable belief of noninfringement. “[I]nduced infringement under § 271(b) requires [*inter alia*] knowledge that the induced acts constitute patent infringement.” *Commil USA, LLC v. Cisco Sys., Inc.*, 575 U.S. 632, 641 (2015), quoting

²⁰ Cisco also sought summary judgment of no pre-suit indirect and willful infringement. This contention is moot considering the court’s ruling that Egenera is not entitled to recover pre-suit damages.

Glob.-Tech Appliances, Inc. v. SEB S.A., 563 U.S. 754, 766 (2011). “[I]f the defendant reads the patent’s claims differently from the plaintiff, and that reading is reasonable,” then the defendant is not liable for indirect infringement. *Id.* at 642. “This knowledge requirement[, however,] may be satisfied under the doctrine of willful blindness. . . . [T]he doctrine of willful blindness requires the patentee to show not only that the accused subjectively believed that there was a high risk of infringement, but also that the accused took deliberate actions to avoid confirming infringement.” *Unwired Planet, LLC v. Apple Inc.*, 829 F.3d 1353, 1364 (Fed. Cir. 2016). Similarly, culpability for willful infringement “is generally measured against the knowledge of the actor at the time of the challenged conduct.” *Halo Elecs., Inc. v. Pulse Elecs., Inc.*, 136 S. Ct. 1923, 1933 (2016). “[A] person is reckless if he acts ‘*knowing or having reason to know* of facts which would lead a reasonable man to realize’ his actions are unreasonably risky.” *Id.* (citation omitted).

In support of its asserted reasonable belief of noninfringement, Cisco relies on the testimony of several of its employees who profess having technical knowledge of the accused UCS system. Each of these witnesses reviewed the claims of the ’430 patent and formed a personal opinion that UCS did not infringe.

Egenera disputes Cisco's reasonableness theory, pointing out that it had explained in detail the basis of its accusations in its infringement contentions, yet none of Cisco's witnesses had considered them before forming their opinions.²¹ The witnesses also did not profess familiarity with the art of performing a patent infringement analysis, and several mistakenly compared UCS to Egenera's BladeFrame product rather than the claims themselves.²² Egenera also argues that the witnesses either did not reference the court's claim construction, or incorrectly applied it. Because intent and willfulness are questions of fact to be determined on consideration of the totality of the circumstances, the court cannot, on this record, find as a matter of law that Cisco held a reasonable belief of noninfringement.

CISCO'S MOTION TO EXCLUDE THE REASONABLE ROYALTY OPINIONS OF DR. SULLIVAN

Cisco seeks to strike the reasonable royalty opinions of Egenera's damages expert, Dr. Ryan Sullivan, on several grounds. Cisco first contends that Dr. Sullivan's royalty calculations were built on a base that is unreliable

²¹ Egenera also contends that Cisco could not have formed a reasonable belief of invalidity because the PTAB had refused to institute Cisco's petition for *inter partes* review of the '430 patent under a lower evidentiary standard on a broader claim construction.

²² Egenera further notes that none of the noninfringement theories provided by the witnesses are advanced by Cisco in support of summary judgment.

because it includes UCS configurations that Egenera does not accuse of infringement. According to Cisco, Dr. Sullivan also failed to discount from the sales of UCS B- and C-series those configurations that do not include fabric extenders or I/O modules. This dispute ultimately is one over fact and not methodology – Egenera does not challenge Cisco’s premise that the royalty base should be limited to the accused products, nor does it accuse UCS configurations that do not include fabric extenders or I/O modules. Relying on the declaration of its expert, Dr. Jones, Egenera maintains that all UCS B-series servers²³ include the I/O module as a required component of the chassis, and that all UCS C-series servers used with UCS Manager incorporate a fabric extender or an I/O module. See Jones Decl. (Dkt # 166-4) ¶ 3. In its brief and reply, Cisco does not point the court to any definitive evidence that B- and C-series servers were sold and/or used in the non-accused configurations.²⁴ On this record, the court cannot conclude that Dr. Sullivan’s royalty base stands on a fatally fictive foundation.

²³ Dr. Sullivan’s analysis does exclude an unaccused variation (Mini) that is not at issue here.

²⁴ Cisco characterizes Dr. Jones’s declaration as an excludable late-disclosed expert opinion. However, whether B- and C-series servers were sold in certain configurations is a matter of fact, and not one of expert interpretation.

Cisco also challenges the acquisition and cost-saving methodologies that Dr. Sullivan used in computing a reasonable royalty. In applying the acquisition approach, Dr. Sullivan looked to what Cisco had paid to acquire Nuova Systems (the original developer of the UCS technology) as a benchmark for what Cisco would be willing to pay to license Egenera's patented technology. Dr. Sullivan first divided the effective acquisition price of Nuova by its revenue to determine the "effective payment share." He then multiplied the "effective payment share" by the UCS per-server revenue to estimate the amount that Cisco would have been willing to pay per-server in exchange for the UCS revenue stream. Finally, Dr. Sullivan applied a "technological apportionment factor" to determine the percentage of the benefit attributable to the patented technology.

Cisco maintains that Dr. Sullivan's calculus is flawed at the multiplication step – in determining the UCS per-server revenue, Dr. Sullivan included the sales of memory and other non-accused items, thereby inflating the per-server revenue figure. Cisco notes that of the ten highest revenue categories that figured in Dr. Sullivan's computation, eight (amounting to 85% of the top ten total) were memory components that Egenera admits do not infringe. Dr. Sullivan's total also included revenue for categories such as replacement batteries, packaging, cable access bars,

plastic panels, cable management rings and straps, rack doors, mounting screws, cage nuts, and sundry other non-accused staple articles.

Egenera counters, and the court agrees for purposes of this motion, that Dr. Sullivan's approach is permitted because the patent is directed to the system as a whole, and not simply a component thereof.²⁵ See, e.g., '430 patent claim 3 (directed to "[a] *platform* for automatically deploying at least one virtual processing area network") (emphasis added). As the Federal Circuit explained in *AstraZeneca AB v. Apotex Corp.*, 782 F.3d 1324 (Fed. Cir. 2015), "it has long been recognized that a patent that combines 'old elements' may 'give[] the entire value to the combination' if the combination itself constitutes a completely new and marketable article." *Id.* at 1338-1339. (citation omitted).

It is not the case that the value of all conventional elements must be subtracted from the value of the patented invention as a whole when assessing damages. For a patent that combines "old elements," removing the value of all of those elements would

²⁵ Contrary to Cisco's suggestion, Dr. Sullivan's analysis is not an application of the so-called "entire market rule." The "entire market rule" provides that when small components of multi-element products are accused of infringement, the patentee may "assess damages based on the entire market value of the accused product only where the patented feature creates the 'basis for customer demand' or 'substantially create[s] the value of the component parts.'" *Uniloc USA, Inc. v. Microsoft Corp.*, 632 F.3d 1292, 1318 (Fed. Cir. 2011) (citation omitted). Because the claims here are directed to a multi-element system, there is no requirement to demonstrate that any particular component of the system drives customer demand.

mean that nothing would remain. In such cases, the question is how much new value is created by the novel combination, beyond the value conferred by the conventional elements alone.

Id. at 1339. Here, the physical components of the claimed platform are not claimed as novel. Rather, the invention resides in the overall arrangement and configuration of the components that are designed to enable the stated function of “deploy[ing a] virtual processing area network.” Accordingly, the court rejects Cisco’s deconstructionist approach in its valuation of the accused UCS system.²⁶

Dr. Sullivan’s cost-saving approach, on the other hand, uses the total-cost-of-ownership (TCO) savings of UCS over competing server deployments as the revenue base. Dr. Sullivan notes that Cisco touts TCO savings as a major benefit of migrating to the UCS system and advertises the amount of savings attributable to the patented technology as “economically equivalent to producer profits in this case.” Egenera Opp’n (Dkt # 165) at 5. To determine the savings attributable to the patented technology, Dr. Sullivan used Cisco’s own online tool to estimate the reduction in customer data center capital and operating expense to be gained by switching to UCS. He then multiplied the per-server savings by both a technical and a

²⁶ Cisco has not presented evidence to suggest that the components whose revenue figured into Dr. Sullivan’s computation were sold separately from UCS.

commercialization apportionment factor to account for the benefit of UCS derived from non-patented technology, and to credit Cisco for its efforts in bringing UCS to market. The resulting value is the per-server royalty rate that informs his model.

The court agrees with Cisco that Dr. Sullivan's cost-saving methodology rests on a jerry-built foundation. The general principle that a lower TCO enables a vendor to charge a customer a premium in the acquisition price is sound. For example, a customer may be willing to pay \$10 for an energy-efficient LED lightbulb instead of \$2 for an incandescent bulb in order to save \$20 in annual electrical costs. It does not follow, however, that the vendor's revenue is equivalent to the customer's TCO.²⁷ The lightbulb maker does not receive the \$20 that the customer saves in electric costs nor does it necessarily earn an equivalent amount on the sale of the LED bulb. So it is the case here. Any premium Cisco charges for the lower TCO feature is already built into the sales price for UCS.²⁸ Dr. Sullivan

²⁷ Egenera cites to Hal Varian's *Intermediate Microeconomics*, 6th ed. (2003), at 388, as evidence of the general acceptance of Dr. Sullivan's methodology. Egenera Opp'n (Dkt # 165) at 13. The textbook explains that a producer's surplus is equal to its revenue less its variable costs (this is the basis of the uncontroversial apportionment steps of Dr. Sullivan's analysis), but the treatise does not equate a customer's TCO savings to a producer's surplus.

²⁸ In the abstract, a customer may be willing to pay up to the TCO savings to achieve a benefit from a bargain. For example, in the game of

offers no evidence or analysis to tie Cisco's revenue to TCO savings other than his *ipse dixit*. Because TCO savings is not a reliable approximation of revenue, the court will exclude Dr. Sullivan's cost-savings analysis.

EGENERA'S MOTION TO EXCLUDE THE REASONABLE ROYALTY OPINIONS OF DR. BECKER

In its turn, Egenera seeks to exclude the reasonable royalty opinions of Cisco's damages expert, Dr. Stephen Becker. Dr. Becker used the valuation of Egenera as a going concern at the time the alleged infringement began as the base for his royalty computation. He excluded portions of the valuation he considered not attributable to the '430 patent (such as servicing of the installed base and foreign sales), then applied a factor equal to Cisco's market share (to reflect the non-exclusive nature of the hypothetical license) and apportioned the value between the patented and non-patented aspects of

Monopoly, a player could elect to pay the bank \$50 to get out of jail or purchase a "get out jail free" card from another player. A rational player would theoretically be willing to pay any amount under \$50 to another player to obtain a savings above and beyond paying the bank to get out of jail. In the reality of the marketplace, however, there are acquisition costs, transaction costs, and other factors at play. (Egenera relies on Cisco's marketing material to suggest that acquisition costs are "almost 'irrelevant.'" See Egenera Opp'n (Dkt # 165) at 12, citing Egenera Ex. 11 (Dkt # 166-11). What Cisco actually said was that "[t]he acquisition cost difference between server vendors is irrelevant" because the costs of servers from different vendors do not vary significantly in the competitive marketplace. See Egenera Ex. 11.) Cisco's sales volume (and thus what Egenera claims as the reasonable royalty base) reflects the marketplace demand for UCS supplied at Cisco's actual (and not some theoretically possible) pricing model.

Egenera's Pan Manager. Finally, Dr. Becker applied the remainder of the *Georgia-Pacific* factors. See *Exmark Mfg. Co. Inc. v. Briggs & Stratton Power Prod. Grp., LLC*, 879 F.3d 1332, 1348-1349 (Fed. Cir. 2018), citing *Georgia-Pacific Corp. v. U.S. Plywood Corp.*, 318 F. Supp. 1116 (S.D.N.Y. 1970).

Egenera asserts that Dr. Becker's methodology is questionable because it does not account for Cisco's use of the '430 patent. See *Aqua Shield v. Inter Pool Cover Team*, 774 F.3d 766, 770 (Fed. Cir. 2014) ("The 'value of what was taken' – the value of the use of the patented technology – measures the royalty.") (citation omitted). Egenera notes that, although Dr. Becker agreed that a patent could be worth more than the company that owns it, he improperly capped the damages at Egenera's market valuation, which did not include the value of Cisco's use of the '430 patent. Further, Egenera characterizes Dr. Becker's application of Cisco's market share to Egenera's market value as "meaningless," Egenera Mot. (Dkt # 144) at 4, because Egenera did not own 100% of the available market, see Egenera Reply (Dkt # 191) at 1.

Cisco responds, and the court agrees, that Dr. Becker's approach is sufficiently plausible. Cisco explains that a patent can be worth more than the company that owns it in a "Rembrandt in the attic" situation in which the

company is not actively practicing the patent. Cisco Opp'n (Dkt # 169) at 5. Here, because Egenera made and sold products embodying the patent, the patent's value, as Dr. Becker saw it, was subsumed in the valuation of Egenera as a company. Dr. Becker's premise, in other words, is not dissimilar from Dr. Sullivan's cost of acquisition approach. Dr. Sullivan looked to what Cisco paid for a company that owned the UCS technology. Dr. Becker treated Egenera's valuation as the amount of money that someone purchasing Egenera, including the '430 patent, would have paid at the time of the infringement. Both approaches begin with the value of the company before making exclusions not attributed to the patented technology. It is for the jury, with the benefit of rigorous cross-examination, to decide the outcome of a reasonable hypothetical negotiation.

ORDER

For the foregoing reasons, Cisco's Motion for Summary Judgment of Unclean Hands is DENIED. Egenera's Motion for Summary Judgment of No Unclean Hands is ALLOWED. Cisco's Motion for Summary Judgment of Noninfringement is ALLOWED-IN-PART as to claims 1 and 5, and otherwise DENIED. Cisco's Motion to Exclude the Infringement Opinions of Dr. Jones is DENIED subject to the caveat to clarify labels. Egenera's Motion for Judgment of No Anticipation is DENIED. Cisco's Motion for

Summary Judgment of No Injunctive Relief is DENIED. Cisco's Motion for Summary Judgment of No Pre-Suit Damages, Indirect Infringement, or Willfulness is ALLOWED-IN-PART as to pre-suit damages, and otherwise DENIED. Cisco's Motion to Exclude the Reasonable Royalty Opinions of Dr. Sullivan is ALLOWED-IN-PART as to the cost-saving analysis, and otherwise DENIED. Egenera's Motion to Exclude the Reasonable Royalty Opinions of Dr. Becker is DENIED. The Clerk will set the remaining claims for trial.

SO ORDERED.

/s/ Richard G. Stearns
UNITED STATES DISTRICT JUDGE

UNITED STATES DISTRICT COURT
DISTRICT OF MASSACHUSETTS

CIVIL ACTION NO. 16-11613-RGS

Egenera, Inc.

v.

CISCO SYSTEMS, INC.

MEMORANDUM AND ORDER ON
MOTION FOR JUDGMENT AS A MATTER OF LAW
AND MOTION FOR A NEW TRIAL

December 15, 2022

STEARNS, D.J.

After a ten-day trial, a jury found that plaintiff Egenera, Inc. (Egenera) had not proven by a preponderance of the evidence that defendant Cisco Systems, Inc.'s (Cisco) Unified Computing System (UCS) product infringes claim 3 or claim 7 of United States Patent No. 7,231,430 (the '430 patent). Egenera seeks to overturn this verdict, moving for judgment as a matter of law pursuant to Fed. R. Civ. P. 50(b), or for a new trial pursuant to Fed. R. Civ. P. 59. For the following reasons, the court will deny both motions.

I. Egenera's Motion for Judgment as a Matter of Law

Egenera argues that it is entitled to judgment as a matter of law because "the jury's verdict cannot be supported without misapplying the law." Mem. in Supp. of Mot. for J. as a Matter of Law (Dkt # 504) at 3.

APPX00077

Specifically, citing *Moba, B.V. v. Diamond Automation, Inc.*, 325 F.3d 1306 (Fed. Cir. 2003), Egenera maintains that the only basis on which the jury could have found non-infringement is by importing additional limitations into the claims.

The court disagrees. The jury was properly instructed to compare the UCS product to the language of the claims themselves (and not to consider the presence or absence of any additional features in assessing infringement), and unlike the situation in *Moba*, the record here provides ample basis to support the jury's findings under the plain language of the claims. For example, with respect to the programming step, Cisco witnesses testified that the central processing unit (CPU) is *not* programmed for the claimed purpose of establishing the specified virtual local area topology – *only* the network interface card (NIC) is.¹ Cisco witnesses similarly testified that the accused product does not practice the modifying or extracting elements for reasons tied directly to the claim language. While Egenera may have found this testimony unpersuasive rebuttal to the testimony of its own

¹ Egenera contends that Cisco misled the jury as to the legal meaning of comprising, improperly leaving the jury with the impression that, because the UCS product programs topology on the NIC, it could not also program topology on the CPU. But the jury was instructed prior to deliberation that comprising means “including the following but not excluding others,” Trial Tr., Day 10 (Dkt # 496) at 193, and Egenera does not cite to any evidence suggesting the jury failed to understand or apply the court's definition.

expert, the fact that the jury did not weigh the evidence as Egenera might have wished does not mean that the jury improperly imported additional limitations from outside the claim language. The court accordingly will deny Egenera's Motion for Judgment as a Matter of Law.

II. Egenera's Motion for a New Trial

Egenera raises four arguments in support of its Motion for a New Trial. Because none of these arguments carry the day, the court will deny the motion.

A. Alleged Violations of the Court's Motion in Limine Rulings

Egenera first contends that Cisco violated the court's rulings on certain motions in limine during closing argument. In light of Egenera's failure to object to the alleged improper arguments,² the court's review is for plain error only.

² Egenera suggests that it was effectively precluded from objecting during the closing by the court's practice of not having sidebars and/or by rules of "professionalism and decorum." Reply in Supp. of Mot. for a New Trial (Dkt #517) at 1. Egenera did not have any difficulty lodging objections at other times during the trial, so the court does not credit the first suggestion. As to the second, even assuming that Egenera felt constrained to not interrupt opposing counsel mid-argument, it does not explain why it did not bring its objections to the court's attention once Cisco's counsel had finished.

Based on its review of the parties' arguments and the record, the court is not convinced that any plain error occurred. The relevant pretrial motion in limine rulings were as follows:

- Testimony, argument, or reference to the absence of any witnesses who do not appear at trial are precluded.
- Cisco is precluded from referring to Egenera as a non-practicing entity or a patent troll.
- The parties are precluded from making any arguments that large companies infringe patents, or that non-practicing entities bring baseless claims. Each party may introduce evidence regarding its own business and the business of the other party.
- ELECTRONIC ORDER entered granting 406 Motion in Limine #7 to exclude references to the parties' ability to finance the current litigation.

July 22, 2022 Order (Dkt # 431); July 21, 2022 Order (Dkt # 412).

As to the first, Mr. Thompson *did* appear at trial, which either places him outside the scope of the ruling or at least provides enough ambiguity that the court cannot deem the alleged transgression as plain error. As to the second, Egenera does not suggest that Cisco expressly called Egenera a “non-practicing entity” or a “patent troll” during its closing argument. It instead contends that Cisco effectively implied that Egenera is a non-practicing entity or patent troll (although does not explain in any convincing way how this implication was conveyed in terms the jury would have understood). The court cannot say that, even if an error occurred, it was plain. As to the

third and fourth rulings, even assuming an error occurred, there is sufficient ambiguity regarding the language used that the court cannot say that any alleged error was plain.

In any event, even if the alleged errors were plain, Egenera has not shown that any of them was prejudicial. The court instructed the jury to compare Cisco's product to the language of the claims, and the jury's questions during the deliberations indicate that it followed these instructions and properly focused on whether Cisco's product met each and every limitation of claims 3 and 7 of the '430 patent. *See* Jury Questions A, B, & C (Dkt # 482). There is no reason to think the jury was improperly swayed by emotive allusions or sotto voce insinuations.

B. Alleged Improper Lay Witness Testimony

Egenera also contends that Cisco elicited improper expert testimony from two of its lay witnesses during trial. But Egenera waived this argument by failing to raise any relevant objection during the examination of either witness.³ Nor can Egenera show prejudice. It opened the door to the

³ An objection was necessary to preserve the challenge. While it is true that the court denied Egenera's motion to exclude the opinions of these witnesses, it did so only because "the opinion testimony is relevant to intent and knowledge for purposes of defending the willful infringement and indirect infringement claims." July 21, 2022 Order (Dkt # 411). The court did not authorize the witnesses to offer expert opinions, so the burden was on Egenera to lodge an *Omega* objection at the appropriate time.

testimony of Mr. Jayakrishnan by asking him about the '430 patent, and by Egenera's own account, much of what Mr. Dvorkin attested to simply "mirrored" the testimony of Mr. Jayakrishnan. Mem. in Supp. of Mot. for a New Trial (Dkt # 507) at 9.

C. Failure to Include a Jury Instruction on Earlier Patents

Egenera next focuses on the court's decision not to include a jury instruction explicitly stating that a product can infringe an earlier patent even if it is also covered by a later patent. It argues that this omission left the jury "unaware that the existence of Cisco's own patents did not preclude or otherwise 'automatically negate infringement.'" *Id.* at 14, quoting *Glaxo Wellcom, Inc v. Andrx Pharms.*, 344 F.3d 1226, 1233 (Fed. Cir. 2003). The court disagrees that the requested instruction was necessary to ensure the jury did not robotically assume non-infringement. Prior to the deliberations, the court instructed the jurors that (1) what mattered for infringement purposes was whether the product met all elements of the claims themselves (*i.e.*, not whether it was subject to any other patents); and (2) the presence

Egenera's midtrial motion to prevent Mr. Jayakrishnan from testifying about demonstrative PX-BJM does not satisfy this requirement. The subject matter of that motion was significantly narrower than what Egenera now asserts, and in any event, the court denied the motion based on waiver and untimeliness. *See* August 10, 2022 Order (Dkt # 468). An untimely objection does not suffice to preserve a challenge to witness testimony.

of additional features would not defeat a showing of infringement. These instructions sufficiently conveyed the fact that a product can infringe an earlier patent even if it is the subject of a later patent.

D. Failure to Include a Curative Jury Instruction on Copying

Egenera lastly maintains that the court's statement to the venire during empanelment that to infringe essentially means to copy without permission, combined with the failure to provide a curative jury instruction, prejudiced the jury to such a degree that a new trial is warranted. The court disagrees that the requested "curative" instruction was necessary. In the first instance, Egenera confuses introductory remarks made to the venire during the winnowing down of prospective jurors with the formal instructions given to the actual jury once seated. With respect to the seated jury, the court did not repeat its generic description of infringement. Rather, the court instructed the jurors, once sworn by the court, that (1) nothing the court said or did during any phase of the trial should influence their ultimate decision on the merits; (2) what mattered for infringement purposes was whether the accused product met all elements of the claims themselves (*i.e.*, not any preferred embodiment); and (3) a party could infringe a patent even if it did not know the patent existed. These instructions were sufficient to convey the concept that literal copying is not required for infringement liability.

ORDER

For the foregoing reasons, Egenera's Motion for Judgment as a Matter of Law is DENIED and Egenera's Motion for a New Trial is also DENIED.

SO ORDERED.

/s/ Richard G. Stearns
UNITED STATES DISTRICT JUDGE

UNITED STATES DISTRICT COURT
DISTRICT OF MASSACHUSETTS

Egenera, Inc.

Plaintiff

v.

CIVIL ACTION 1:16-11613-RGS

Cisco Systems, Inc.

Defendant

JUDGMENT

STEARNS, D.J.

Consistent with the verdict of the jury, judgment shall enter for defendant Cisco Systems, Inc., on plaintiff Egenera, Inc.'s claims of infringement of the asserted claims of the '430 patent. Judgment shall enter for Egenera on Cisco's claim that the asserted claims of the '430 patent are invalid.

By the court,

December 16, 2022
Date

/s/ Arnold Pacheco
Deputy Clerk

APPX00085



US007231430B2

(12) **United States Patent**
Brownell et al.

(10) **Patent No.:** **US 7,231,430 B2**
(45) **Date of Patent:** **Jun. 12, 2007**

(54) **RECONFIGURABLE, VIRTUAL
PROCESSING SYSTEM, CLUSTER,
NETWORK AND METHOD**

(75) Inventors: **Vern Brownell**, Chatham, MA (US);
Pete Manca, Sterling, MA (US); **Ben
Sprachman**, Hopkinton, MA (US);
Paul Curtis, Sudbury, MA (US); **Ewan
Milne**, Stow, MA (US); **Max Smith**,
Natick, MA (US); **Alan Greenspan**,
Northboro, MA (US); **Scott Geng**,
Westboro, MA (US); **Dan Busby**,
Sterling, MA (US); **Edward Duffy**,
Arlington, MA (US); **Peter Schulter**,
Hampstead, NH (US)

(73) Assignee: **Egenera, Inc.**, Marlboro, MA (US)

(*) Notice: Subject to any disclaimer, the term of this
patent is extended or adjusted under 35
U.S.C. 154(b) by 876 days.

(21) Appl. No.: **10/038,353**

(22) Filed: **Jan. 4, 2002**

(65) **Prior Publication Data**
US 2003/0130833 A1 Jul. 10, 2003

Related U.S. Application Data
(60) Provisional application No. 60/285,296, filed on Apr.
20, 2001.
(51) **Int. Cl.**
G06F 15/16 (2006.01)
G06F 15/173 (2006.01)
G06F 15/167 (2006.01)
H04L 12/56 (2006.01)
(52) **U.S. Cl.** **709/218; 709/212; 709/205;**
709/244; 370/395.53
(58) **Field of Classification Search** **709/204;**
709/205, 208, 212, 216, 223, 238, 244, 245;
700/99; 712/98; 370/254, 395.53

See application file for complete search history.

(56) **References Cited**

U.S. PATENT DOCUMENTS

5,208,811 A	5/1993	Kashio et al.
5,473,599 A	12/1995	Li et al.
5,535,338 A	7/1996	Krause et al.
5,546,535 A *	8/1996	Stallmo et al. 714/9
5,590,285 A	12/1996	Krause et al.
5,818,842 A	10/1998	Burwell et al.
5,825,772 A	10/1998	Dobbins et al.
5,835,725 A	11/1998	Chiang et al.

(Continued)

FOREIGN PATENT DOCUMENTS

WO WO02086712 A1 * 10/2002

Primary Examiner—Bunjoo Jaroenchonwanit

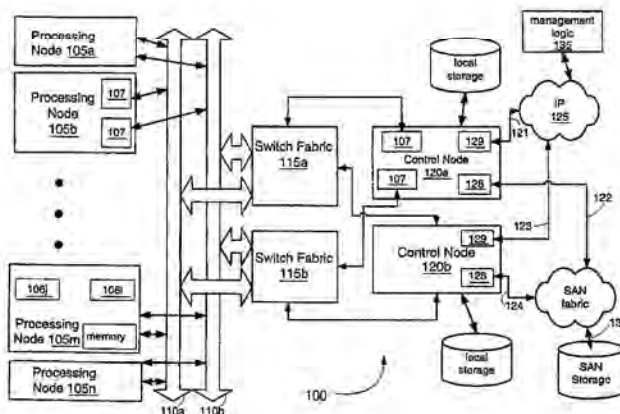
Assistant Examiner—Ramsey Refai

(74) *Attorney, Agent, or Firm*—Wilmer Cutler Pickering
Hale and Dorr LLP

(57) **ABSTRACT**

A platform and method of deploying virtual processing areas
networks are described. A plurality of computer processors
are connected to an internal communication network. At
least one control node is in communication with an external
communication network and an external storage network has
an external storage address space. The at least one control
node is connected to the internal network and thereby is in
communication with the plurality of computer processors.
Configuration logic defines and establishes a virtual pro-
cessing area network having a corresponding set of com-
puter processors from the plurality of processors, a virtual
local area communication network providing communica-
tion among the set of computer processors, and a virtual
storage space with a defined correspondence to the address
space of the storage network.

8 Claims, 14 Drawing Sheets



US 7,231,430 B2

Page 2

U.S. PATENT DOCUMENTS

5,970,066 A	10/1999	Lowry et al.			
6,003,137 A	12/1999	Kawasaki			
6,091,732 A	7/2000	Alexander, Jr. et al.			
6,148,414 A *	11/2000	Brown et al.	714/9		
6,178,171 B1	1/2001	Alexander, Jr. et al.			
6,195,705 B1	2/2001	Leung			
6,411,625 B1 *	6/2002	Furubashi et al.	370/395.53		
6,480,901 B1 *	11/2002	Weber et al.	709/246		
6,597,956 B1 *	7/2003	Aziz et al.	700/3		
6,640,278 B1 *	10/2003	Nolan et al.	711/6		
6,662,221 B1 *	12/2003	Gonda et al.	709/223		
6,675,268 B1 *	1/2004	DeKoning et al.	711/151		
6,701,358 B1 *	3/2004	Poisson et al.	709/223		
6,714,980 B1 *	3/2004	Markson et al.	709/226		
6,757,753 B1 *	6/2004	DeKoning et al.	710/38		
6,779,016 B1 *	8/2004	Aziz et al.	709/201		
6,789,090 B1 *	9/2004	Miyake et al.	707/104.1		
6,820,171 B1 *	11/2004	Weber et al.	711/114		
6,883,065 B1 *	4/2005	Pittelkow et al.	711/114		
6,950,871 B1 *	9/2005	Honma et al.	709/226		
6,971,044 B2 *	11/2005	Geng et al.	714/11		
7,174,390 B2 *	2/2007	Schulter et al.	709/245		
7,188,062 B1 *	3/2007	Rieschl et al.	703/23		

* cited by examiner

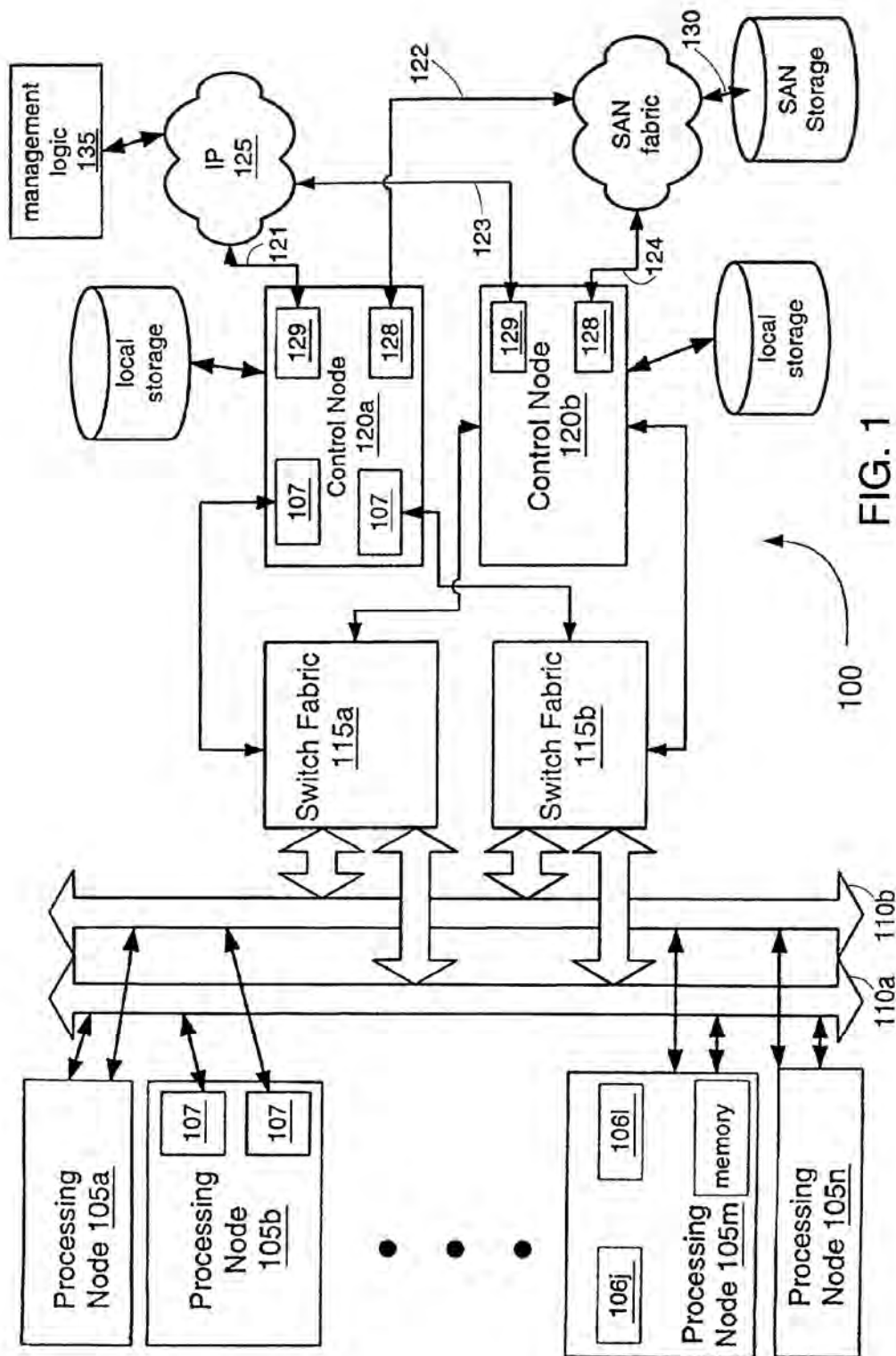


FIG. 1

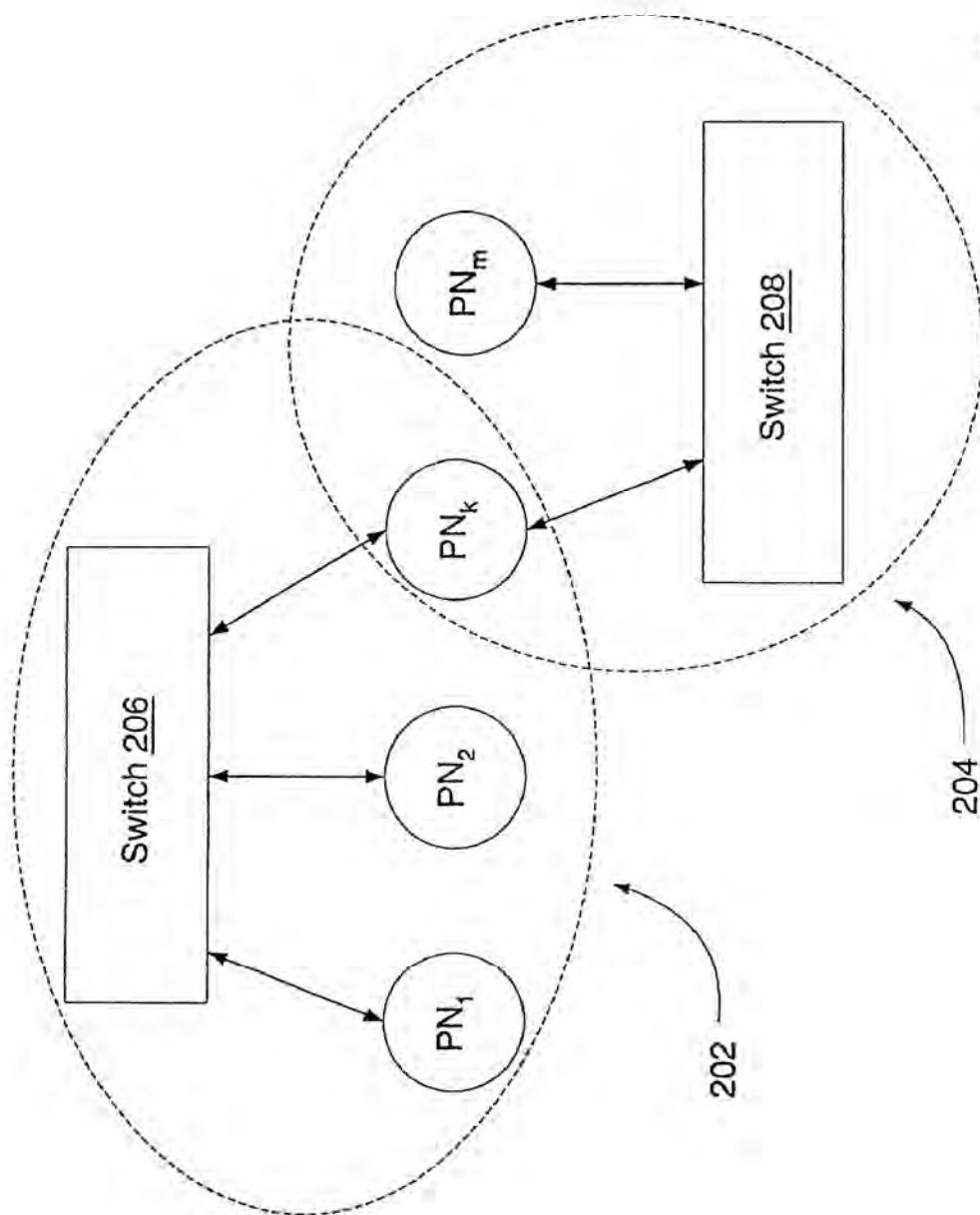


FIG. 2A

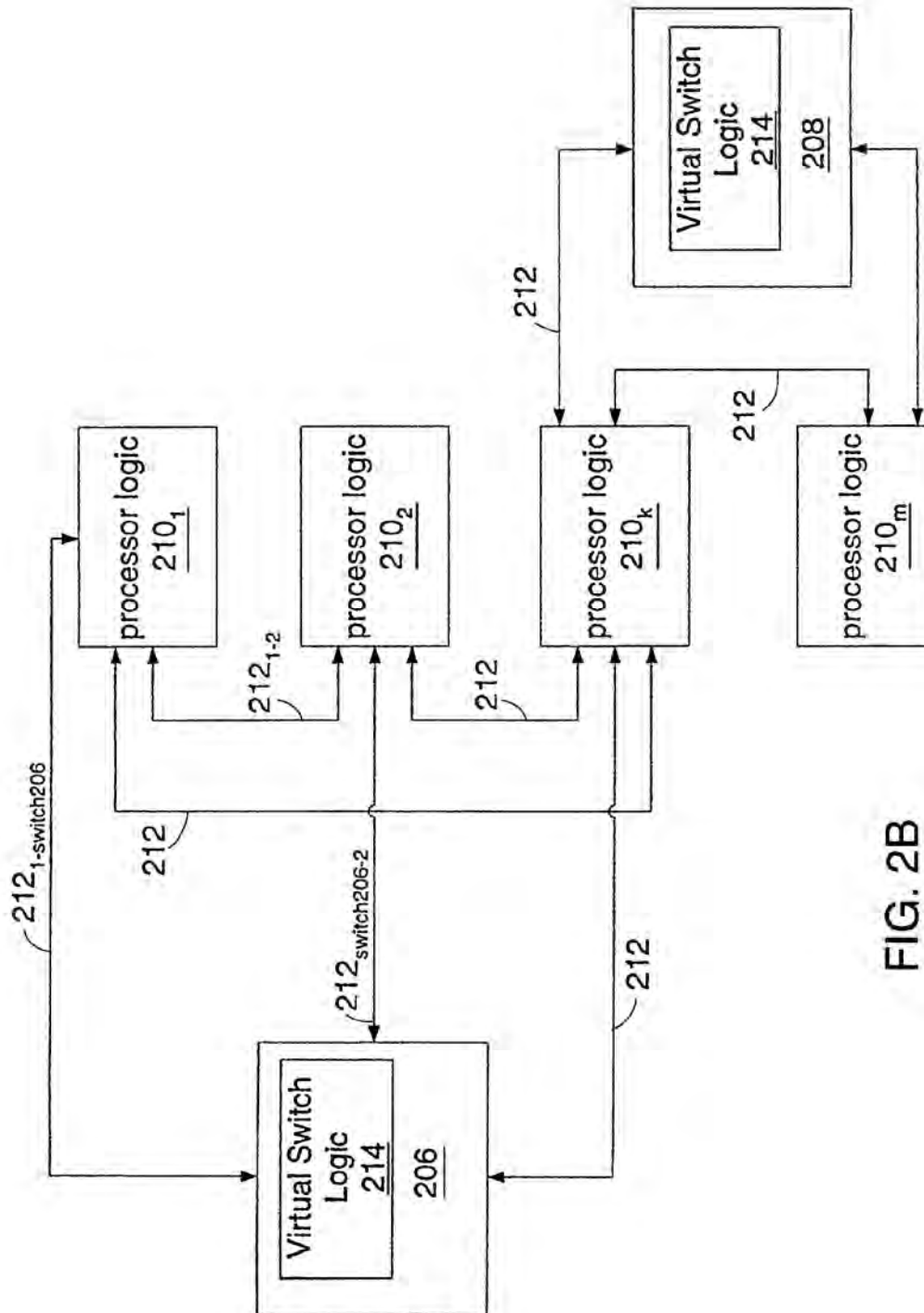


FIG. 2B

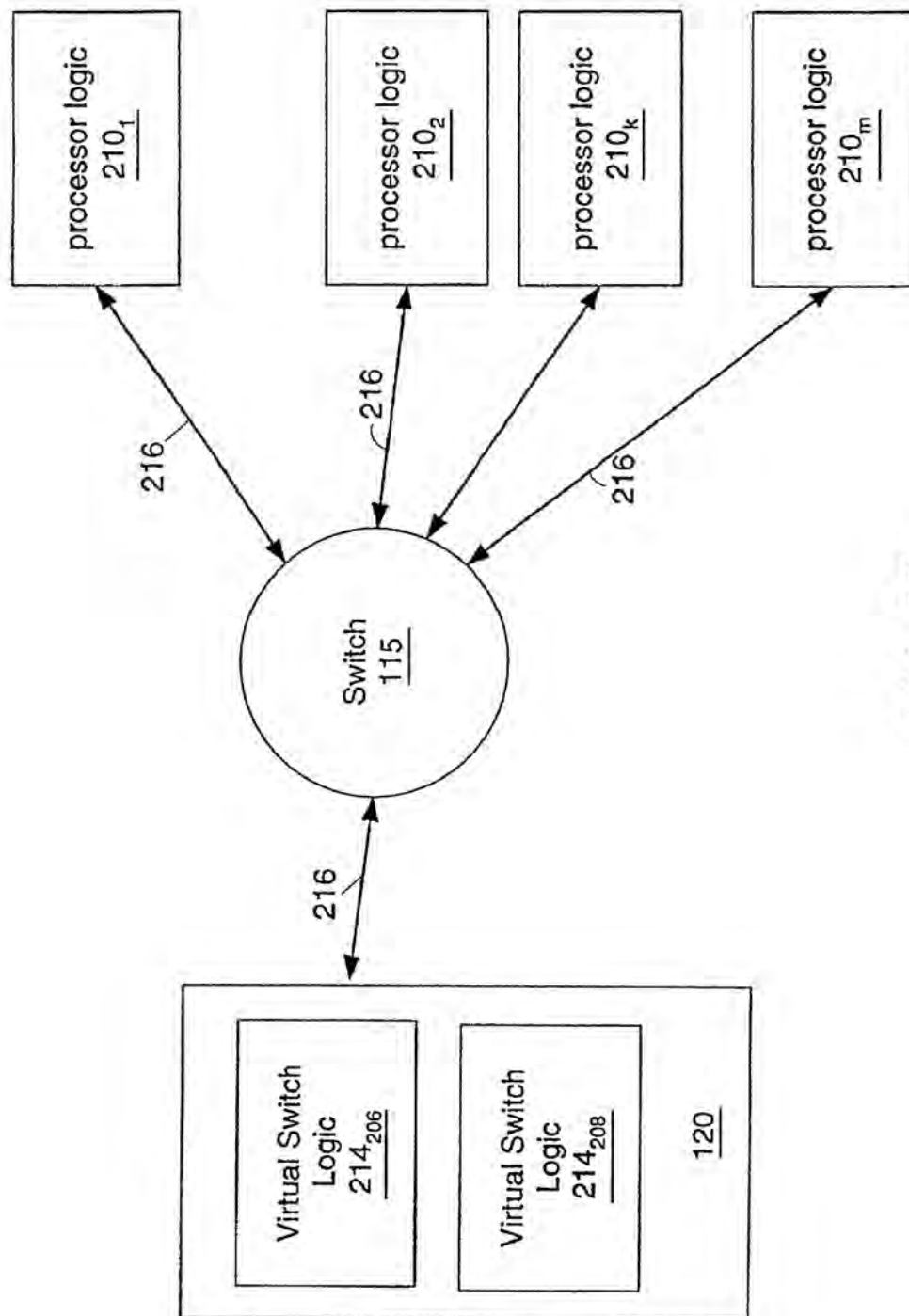


FIG. 2C

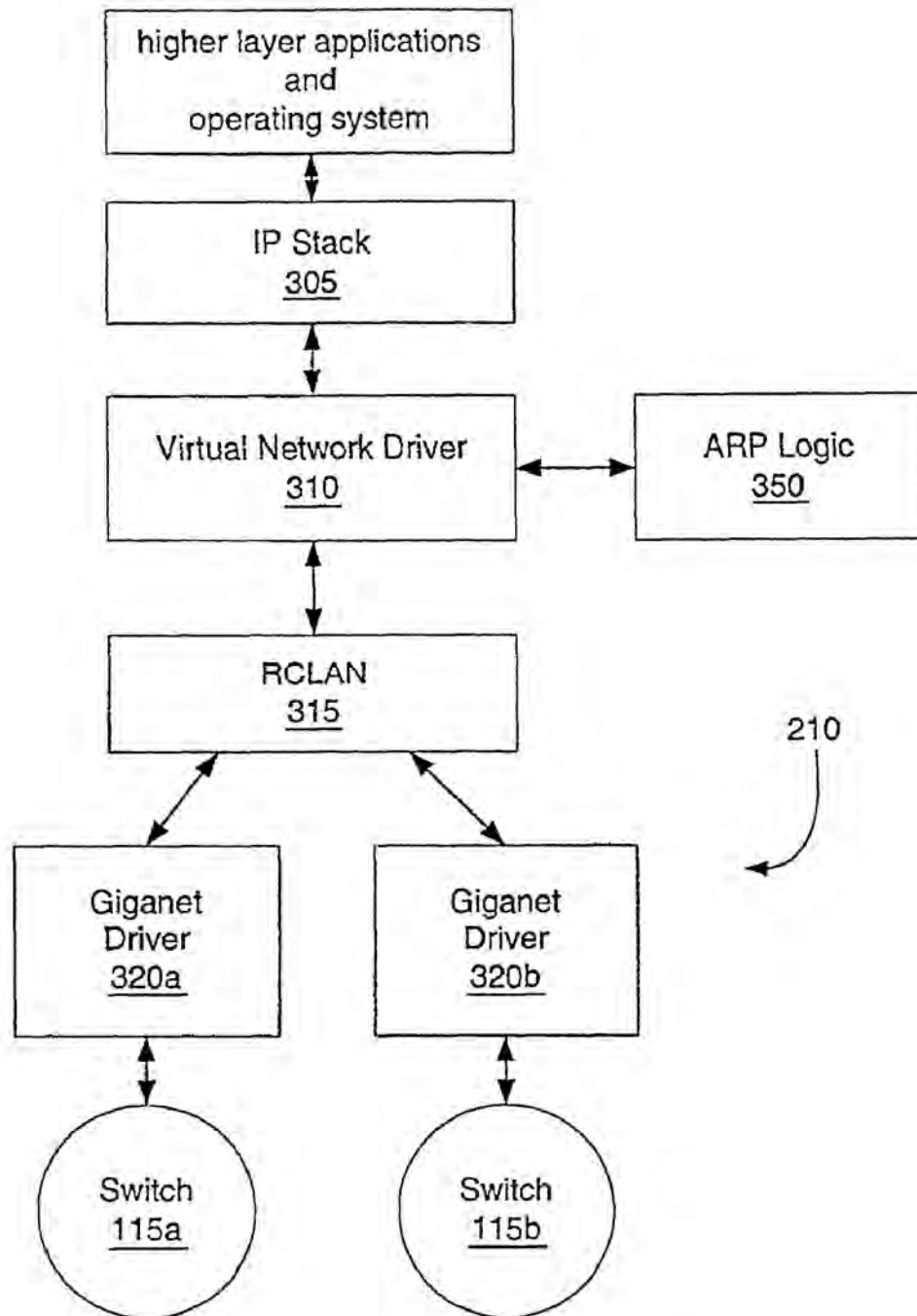


FIG. 3A

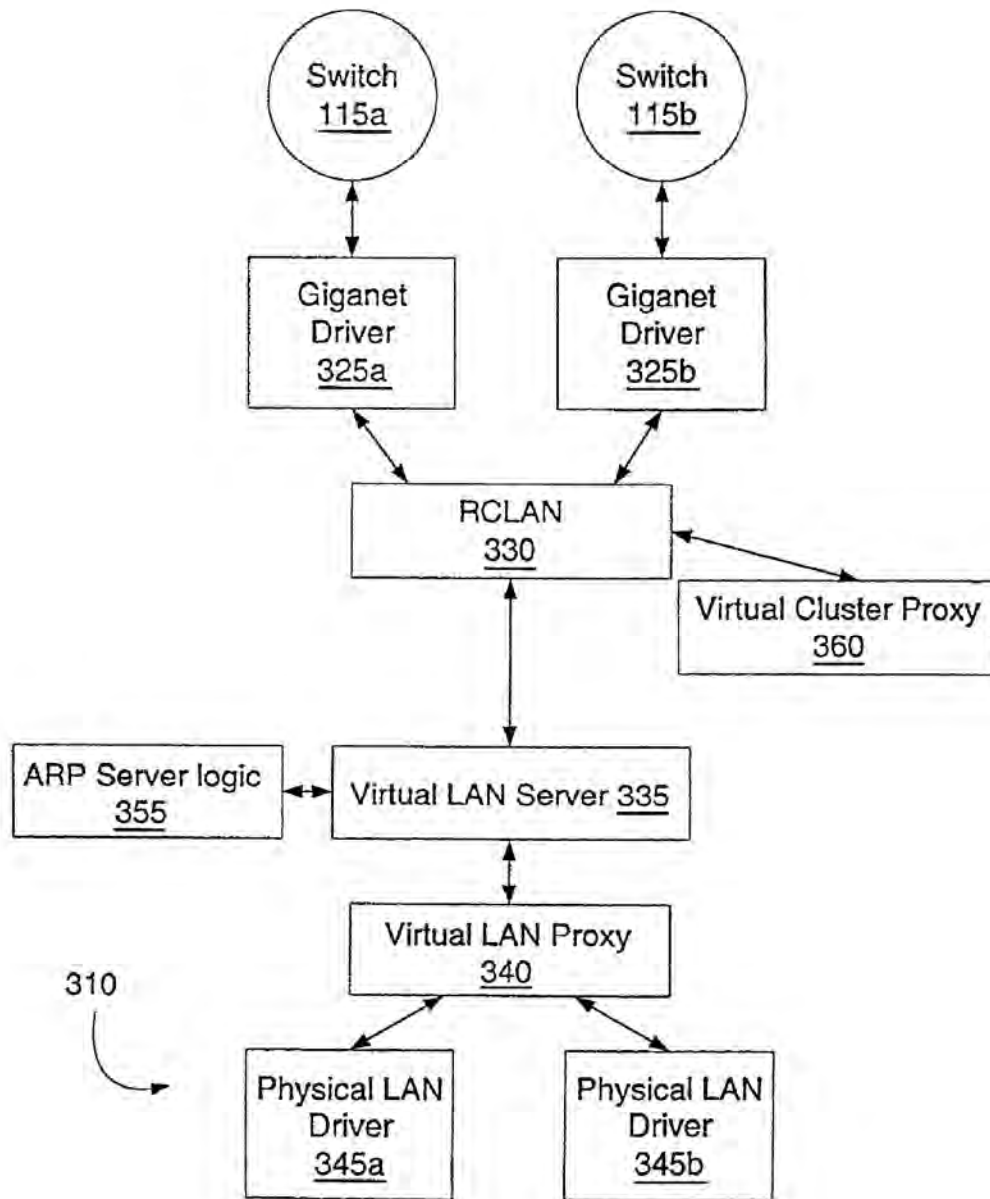


FIG. 3B

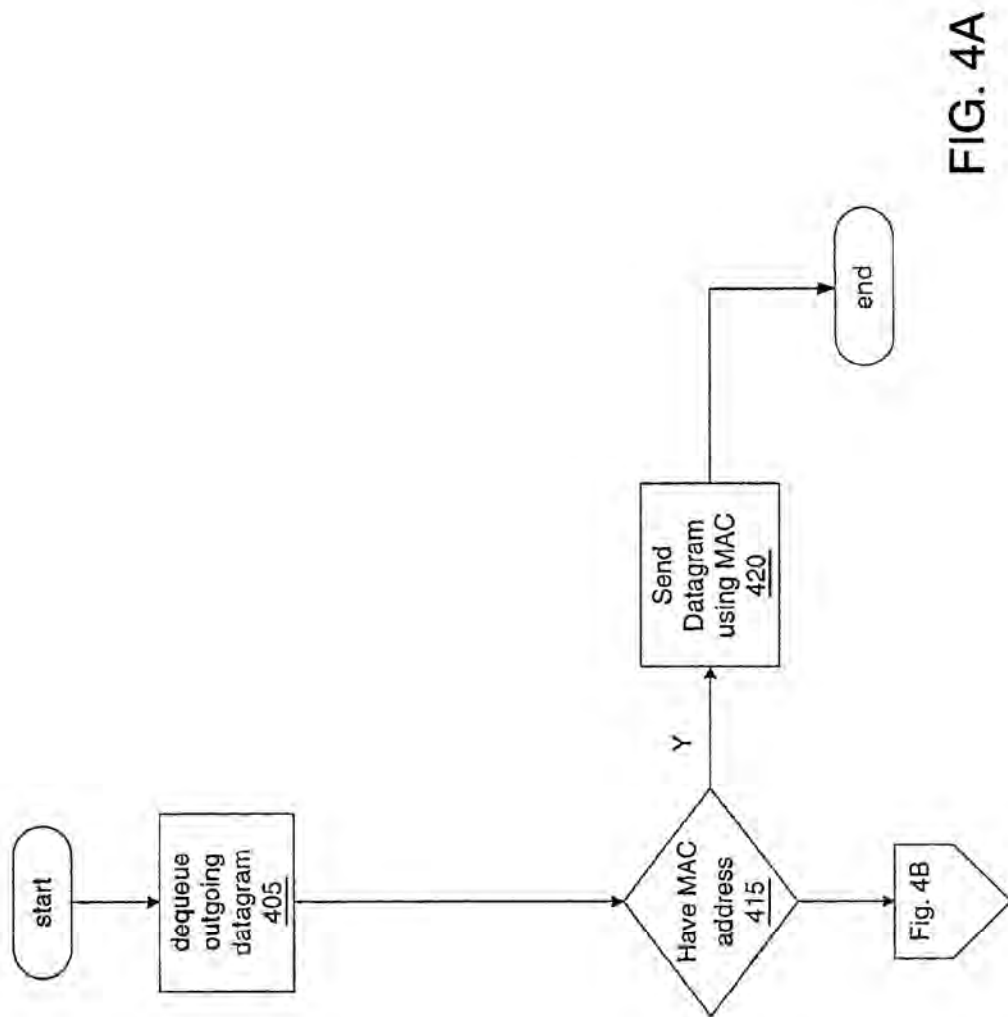


FIG. 4A

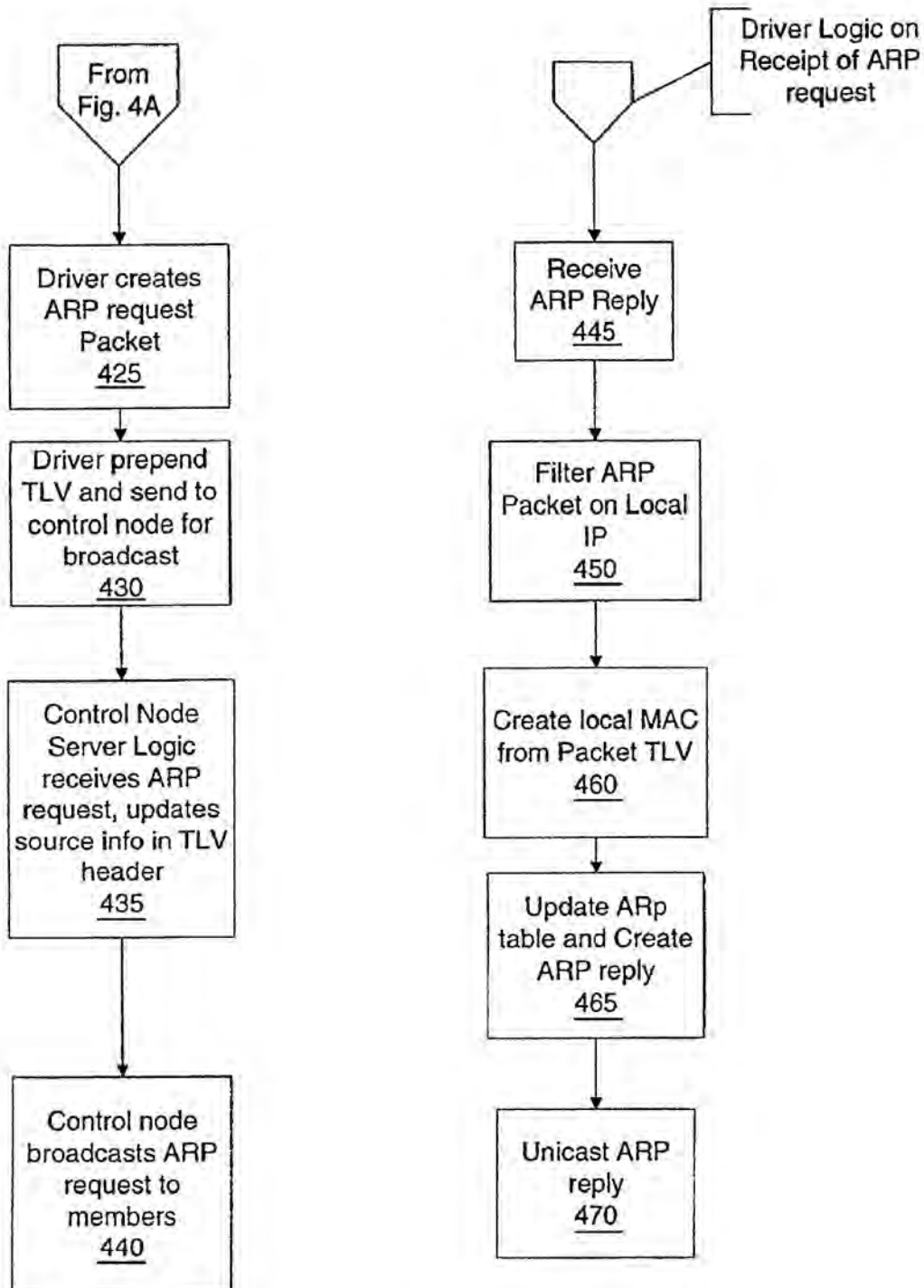


FIG. 4B

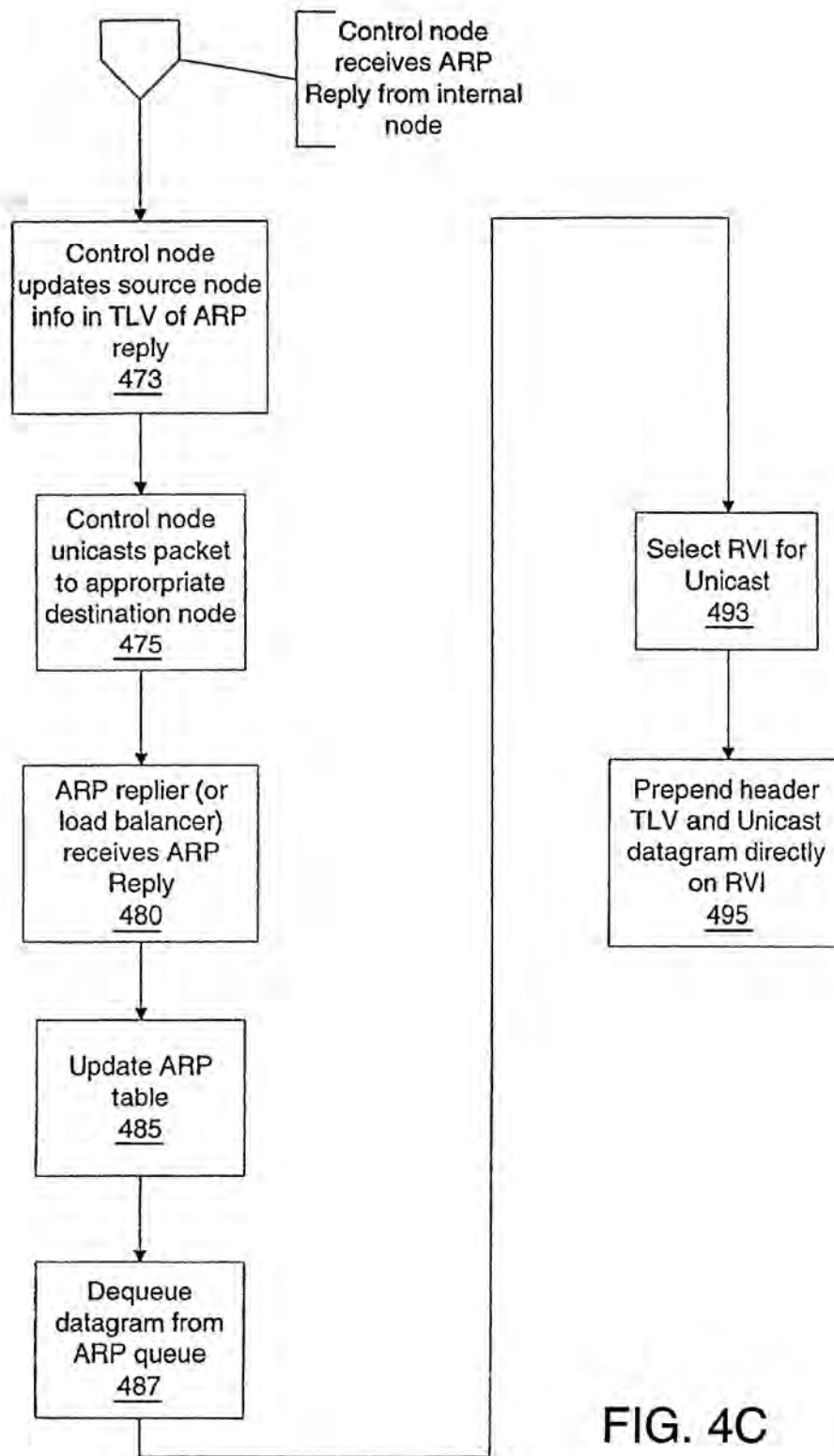


FIG. 4C

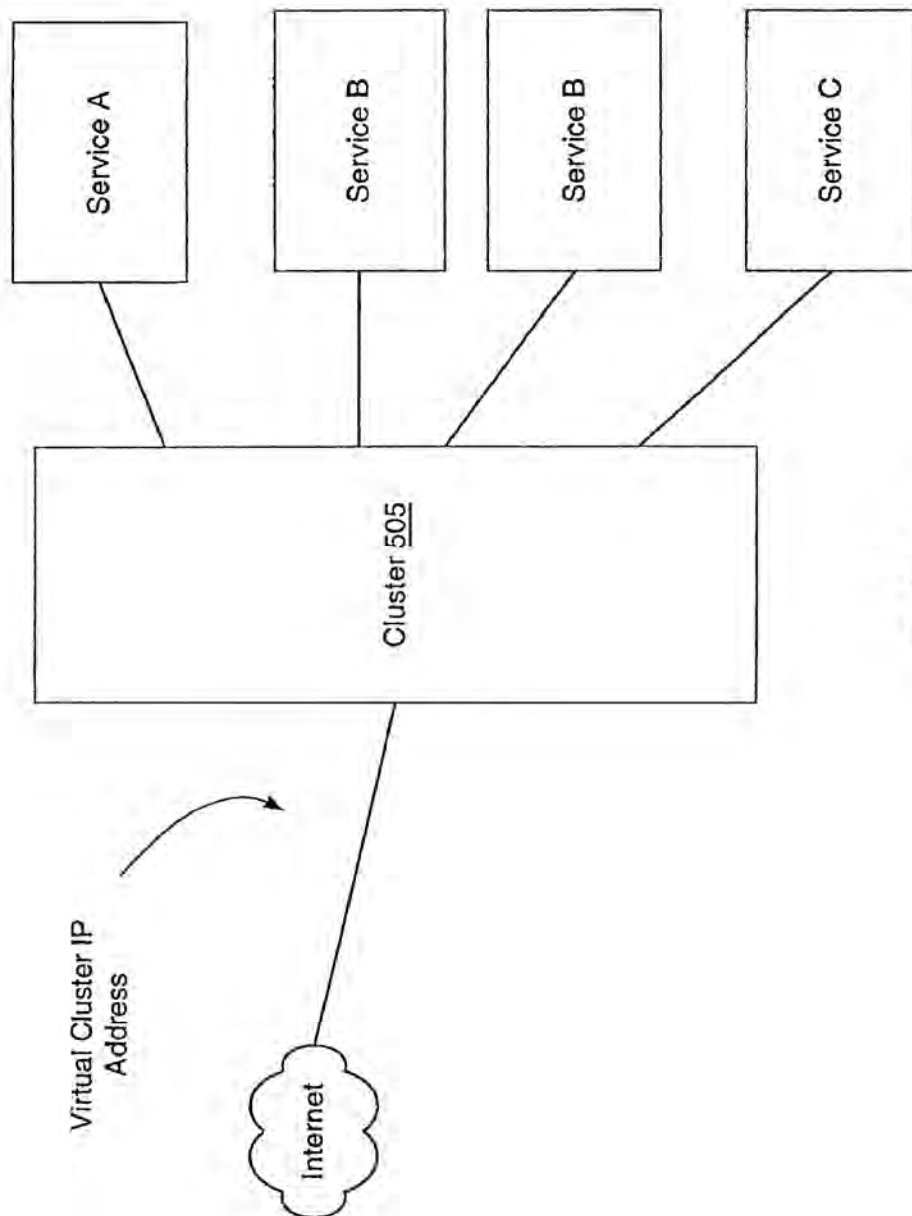


FIG. 5

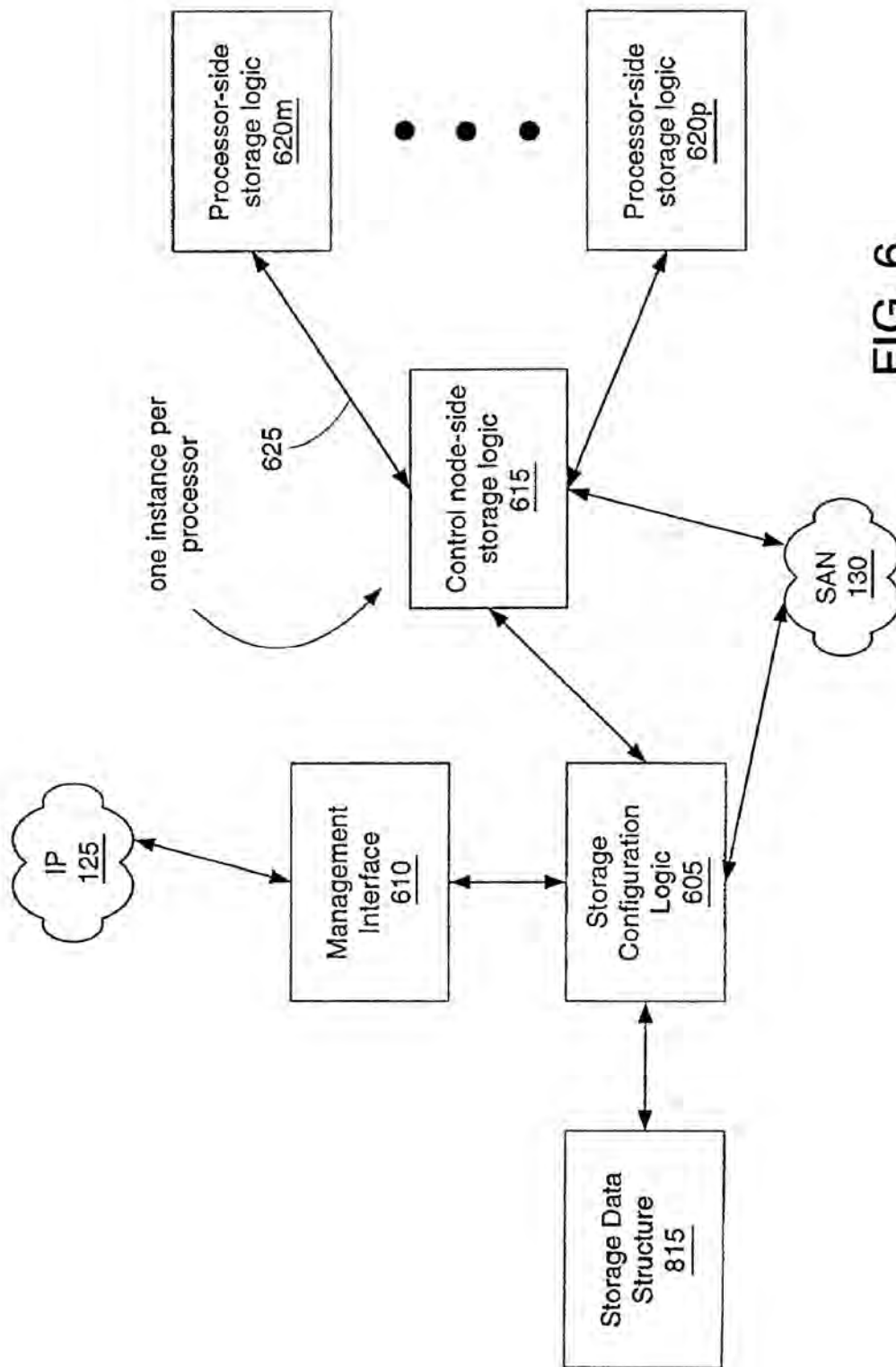
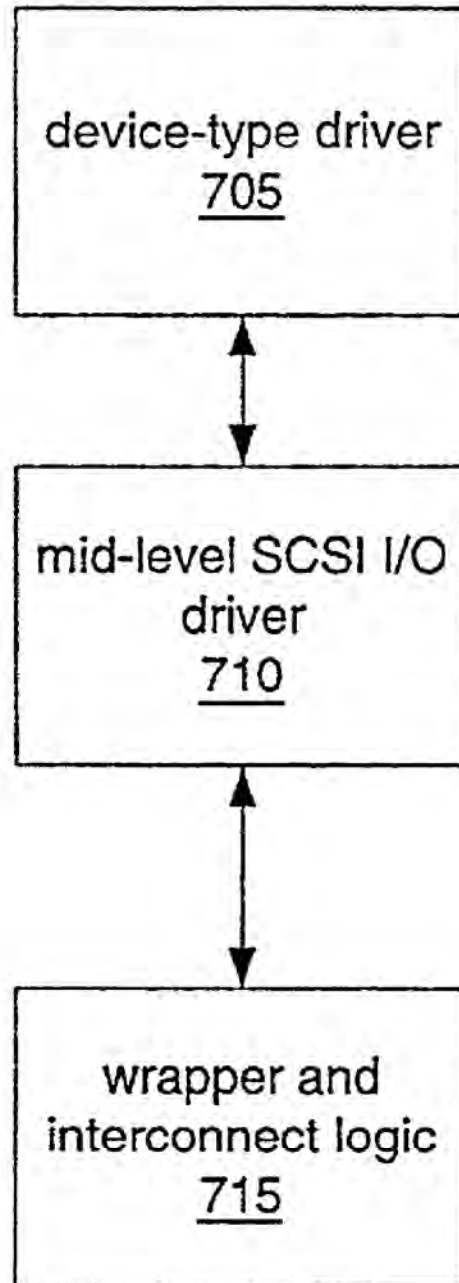


FIG. 6

**FIG. 7**

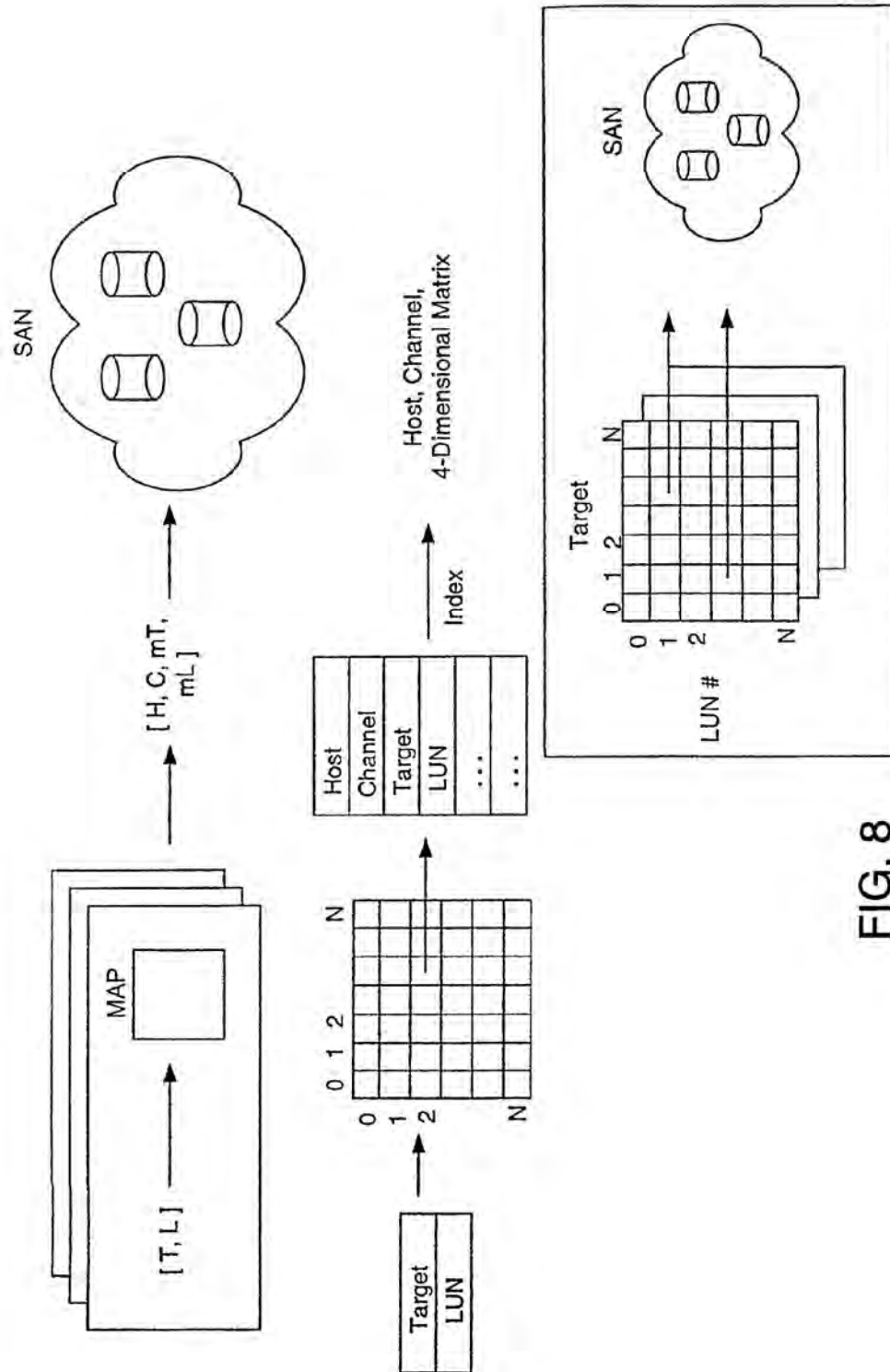


FIG. 8

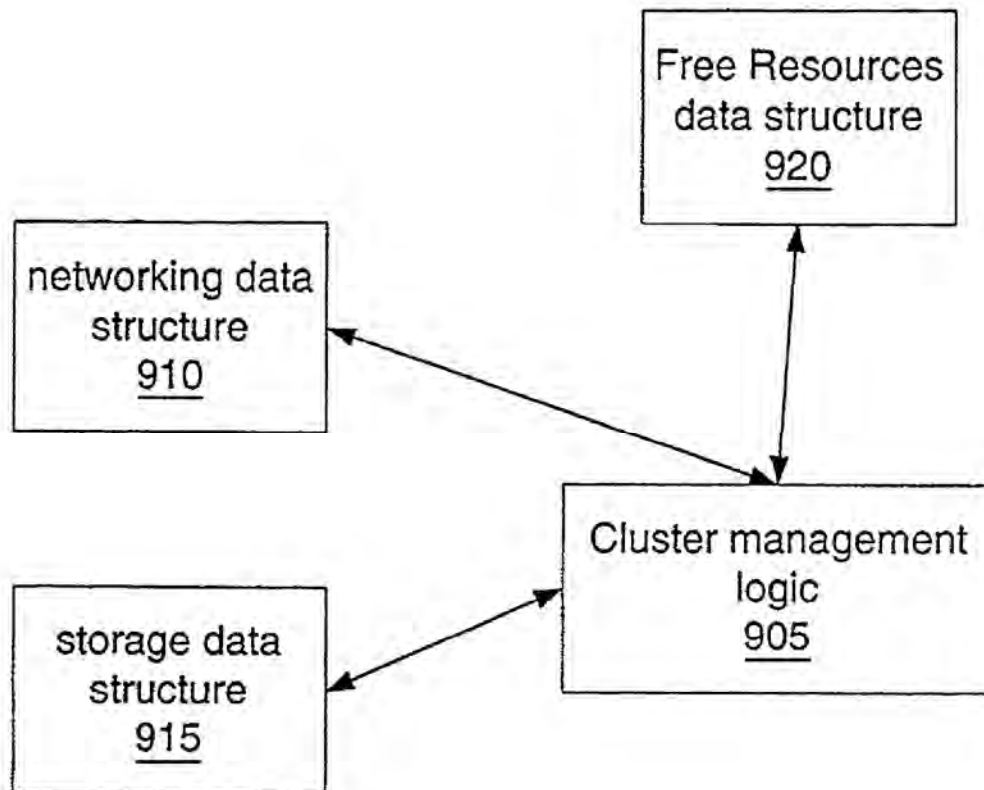


FIG. 9

RECONFIGURABLE, VIRTUAL PROCESSING SYSTEM, CLUSTER, NETWORK AND METHOD

CROSS-REFERENCE TO RELATED APPLICATION

This application claims priority to U.S. provisional application Ser. No. 60/285,296, filed on Apr. 20, 2001, which is hereby incorporated by reference.

BACKGROUND

1. Field of the Invention

The present invention relates to computing systems for enterprises and application service providers and, more specifically, to processing systems having virtualized communication networks and storage for quick deployment and reconfiguration.

2. Discussion of Related Art

In current enterprise computing and application service provider environments, personnel from multiple information technology (IT) functions (electrical, networking, etc.) must participate to deploy processing and networking resources. Consequently, because of scheduling and other difficulties in coordinating activities from multiple departments, it can take weeks or months to deploy a new computer server. This lengthy, manual process increases both human and equipment costs, and delays the launch of applications.

Moreover, because it is difficult to anticipate how much processing power applications will require, managers typically over-provision the amount of computational power. As a result, data-center computing resources often go unutilized or under-utilized.

If more processing power is eventually needed than originally provisioned, the various IT functions will again need to coordinate activities to deploy more or improved servers, connect them to the communication and storage networks and so forth. This task gets increasingly difficult as the systems become larger.

Deployment is also problematic. For example, when deploying 24 conventional servers, more than 100 discrete connections may be required to configure the overall system. Managing these cables is an ongoing challenge, and each represents a failure point. Attempting to mitigate the risk of failure by adding redundancy can double the cabling, exacerbating the problem while increasing complexity and costs.

Provisioning for high availability with today's technology is a difficult and costly proposition. Generally, a failover server must be deployed for every primary server. In addition, complex management software and professional services are usually required.

Generally, it is not possible to adjust the processing power or upgrade the CPUs on a legacy server. Instead, scaling processor capacity and/or migrating to a vendor's next-generation architecture often requires a "forklift upgrade," meaning more hardware/software systems are added, needing new connections and the like.

Consequently, there is a need for a system and method of providing a platform for enterprise and ASP computing that addresses the above shortcomings.

SUMMARY

The present invention features a platform and method for computer processing in which virtual processing area networks may be configured and deployed.

According to one aspect of the invention, a computer processing platform includes a plurality of computer processors connected to an internal communication network. At least one control node is in communication with an external communication network and an external storage network having an external storage address space. The at least one control node is connected to the internal network and thereby communicates with the plurality of computer processors. Configuration logic defines and establishes a virtual processing area network having a corresponding set of computer processors from the plurality of processors, a virtual local area communication network providing communication among the set of computer processors but excluding the processors from the plurality not in the defined set, and a virtual storage space with a defined correspondence to the address space of the storage network.

BRIEF DESCRIPTION OF THE DRAWINGS

In the Drawing,

FIG. 1 is a system diagram illustrating one embodiment of the invention;

FIGS. 2A-C are diagrams illustrating the communication links established according to one embodiment of the invention;

FIGS. 3A-B are diagrams illustrating the networking software architecture of certain embodiments of the invention;

FIGS. 4A-C are flowcharts illustrating driver logic according to certain embodiments of the invention;

FIG. 5 illustrates service clusters according to certain embodiments of the invention;

FIG. 6 illustrates the storage software architecture of certain embodiments of the invention;

FIG. 7 illustrates the processor-side storage logic of certain embodiments of the invention;

FIG. 8 illustrates the storage address mapping logic of certain embodiments of the invention; and

FIG. 9 illustrates the cluster management logic of certain embodiments of the invention.

DETAILED DESCRIPTION

Preferred embodiments of the invention provide a processing platform from which virtual systems may be deployed through configuration commands. The platform provides a large pool of processors from which a subset may be selected and configured through software commands to form a virtualized network of computers ("processing area network" or "processor clusters") that may be deployed to serve a given set of applications or customer. The virtualized processing area network (PAN) may then be used to execute customer specific applications, such as web-based server applications. The virtualization may include virtualization of local area networks (LANs) or the virtualization of I/O storage. By providing such a platform, processing resources may be deployed rapidly and easily through software via configuration commands, e.g., from an administrator, rather than through physically providing servers, cabling network and storage connections, providing power to each server and so forth.

Overview of the Platform and its Behavior

As shown in FIG. 1, a preferred hardware platform 100 includes a set of processing nodes 105a-n connected to a switch fabrics 115a,b via high-speed, interconnect 110a,b.

The switch fabric **115a,b** is also connected to at least one control node **120a,b** that is in communication with an external IP network **125** (or other data communication network), and with a storage area network (SAN) **130**. A management application **135**, for example, executing remotely, may access one or more of the control nodes via the IP network **125** to assist in configuring the platform **100** and deploying virtualized PANs.

Under certain embodiments, about 24 processing nodes **105a-n**, two control nodes **120**, and two switch fabrics **115a,b** are contained in a single chassis and interconnected with a fixed, pre-wired mesh of point-to-point (PtP) links. Each processing node **105** is a board that includes one or more (e.g., 4) processors **106j-l**, one or more network interface cards (NICs) **107**, and local memory (e.g., greater than 4 Gbytes) that, among other things, includes some BIOS firmware for booting and initialization. There is no local disk for the processors **106**; instead all storage, including storage needed for paging, is handled by SAN storage devices **130**.

Each control node **120** is a single board that includes one or more (e.g., 4) processors, local memory, and local disk storage for holding independent copies of the boot image and initial file system that is used to boot operating system software for the processing nodes **105** and for the control nodes **106**. Each control node communicates with SAN **130** via 100 megabyte/second fibre channel adapter cards **128** connected to fibre channel links **122**, **124** and communicates with the Internet (or any other external network) **125** via an external network interface **129** having one or more Gigabit Ethernet NICs connected to Gigabit Ethernet links **121**, **123**. (Many other techniques and hardware may be used for SAN and external network connectivity.) Each control node includes a low speed Ethernet port (not shown) as a dedicated management port, which may be used instead of remote, web-based management via management application **135**.

The switch fabrics is composed of one or more 30-port Gigaset switches **115**, such as the NIC-CLAN **1000** and clan **5300** switch, and the various processing and control nodes use corresponding NICs for communication with such a fabric module. Gigaset switch fabrics have the semantics of a Non-Broadcast Multiple Access (NBMA) network. All inter-node communication is via a switch fabric. Each link is formed as a serial connection between a NIC **107** and a port in the switch fabric **115**. Each link operates at 112 megabytes/second.

In some embodiments, multiple cabinets or chassis may be connected together to form larger platforms. And in other embodiments the configuration may differ; for example, redundant connections, switches and control nodes may be eliminated.

Under software control, the platform supports multiple, simultaneous and independent processing areas networks (PANs). Each PAN, through software commands, is configured to have a corresponding subset of processors **106** that may communicate via a virtual local area network that is emulated over the PtP mesh. Each PAN is also configured to have a corresponding virtual I/O subsystem. No physical deployment or cabling is needed to establish a PAN. Under certain preferred embodiments, software logic executing on the processor nodes and/or the control nodes emulates switched Ethernet semantics; other software logic executing on the processor nodes and/or the control nodes provides virtual storage subsystem functionality that follows SCSI semantics and that provides independent I/O address spaces for each PAN.

Network Architecture

Certain preferred embodiments allow an administrator to build virtual, emulated LANs using virtual components, interfaces, and connections. Each of the virtual LANs can be internal and private to the platform **100**, or multiple processors may be formed into a processor cluster externally visible as a single IP address.

Under certain embodiments, the virtual networks so created emulate a switched Ethernet network, though the physical, underlying network is a PtP mesh. The virtual network utilizes IEEE MAC addresses, and the processing nodes support IETF ARP processing to identify and associate IP addresses with MAC addresses. Consequently, a given processor node replies to an ARP request consistently whether the ARP request came from a node internal or external to the platform.

FIG. 2A shows an exemplary network arrangement that may be modeled or emulated. A first subnet **202** is formed by processing nodes PN_1 , PN_2 , and PN_k that may communicate with one another via switch **206**. A second subnet **204** is formed by processing nodes PN_k and PN_m that may communicate with one another via switch **208**. Under switched Ethernet semantics, one node on a subnet may communicate directly with another node on the subnet; for example, PN_1 may send a message to PN_2 . The semantics also allow one node to communicate with a set of the other nodes; for example PN_1 may send a broadcast message to other nodes. The processing nodes PN_1 and PN_2 cannot directly communicate with PN_m because PN_m is on a different subnet. For PN_1 and PN_2 to communicate with PN_m higher layer networking software would need to be utilized, which software would have a fuller understanding of both subnets. Though not shown in the figure, a given switch may communicate via an "uplink" to another switch or the like. As will be appreciated given the description below, the need for such uplinks is different than their need when the switches are physical. Specifically, since the switches are virtual and modeled in software they may scale horizontally as wide as needed. (In contrast, physical switches have a fixed number of physical ports sometimes the uplinks are needed to provide horizontal scalability.)

FIG. 2B shows exemplary software communication paths and logic used under certain embodiments to model the subnets **202** and **204** of FIG. 2A. The communication paths **212** connect processing nodes PN_1 , PN_2 , PN_k , and PN_m , specifically their corresponding processorside network communication logic **210**, and they also connect processing nodes to control nodes. (Though drawn as a single instance of logic for the purpose of clarity, PN_k may have multiple instances of the corresponding processor logic, one per subnet, for example.) Under preferred embodiments, management logic and the control node logic are responsible for establishing, managing and destroying the communication paths. The individual processing nodes are not permitted to establish such paths.

As will be explained in detail below, the processor logic and the control node logic together emulate switched Ethernet semantics over such communication paths. For example, the control nodes have control node-side virtual switch logic **214** to emulate some (but not necessarily all) of the semantics of an Ethernet switch, and the processor logic includes logic to emulate some (but not necessarily all) of the semantics of an Ethernet driver.

Within a subnet, one processor node may communicate directly with another via a corresponding virtual interface **212**. Likewise, a processor node may communicate with the control node logic via a separate virtual interface. Under

certain embodiments, the underlying switch fabric and associated logic (e.g., switch fabric manager logic, not shown) provides the ability to establish and manage such virtual interfaces (VIs) over the point to point mesh. Moreover, these virtual interfaces may be established in a reliable, redundant fashion and are referred to herein in as RVIs. At points in this description, the terms virtual interface (VI) and reliable virtual interface (RVI) are used interchangeably, as the choice between a VI versus an RVI largely depends on the amount of reliability desired by the system at the expense of system resources.

Referring conjointly to FIGS. 2A-B, if node PN_1 is to communicate with node PN_2 , it does so ordinarily by virtual interface 212_{1-2} . However, preferred embodiments allow communication between PN_1 and PN_2 to occur via switch emulation logic, if for example VI 212_{1-2} is not operating satisfactorily. In this case a message may be sent via VI $212_{1-processor106}$ and via VI $212_{switch206-2}$. If PN_1 is to broadcast or multicast a message to other nodes in the subnet 202 it does so by sending the message to control node-side logic 214 via virtual interface $212_{switch206}$. Control node-side logic 214 then emulates the broadcast or multicast functionality by cloning and sending the message to the other relevant nodes using the relevant VIs. The same or analogous VIs may be used to convey other messages requiring control node-side logic. For example, as will be described below, control node-side logic includes logic to support the address resolution protocol (ARP), and VIs are used to communicate ARP replies and requests to the control node. Though the above description suggests just one VI between processor logic and control logic, many embodiments employ several such connections. Moreover, though the figures suggest symmetry in the software communication paths, the architecture actually allows asymmetric communication. For example, as will be discussed below, for communication clustered services the packets would be routed via the control node. However, return communication may be direct between nodes.

Notice that like the network of FIG. 2A, there is no mechanism for communication between node PN_2 , and PN_m . Moreover, by having communication paths managed and created centrally (instead of via the processing nodes) such a path is not creatable by the processing nodes, and the defined subnet connectivity cannot be violated by a processor.

FIG. 2C shows the exemplary physical connections of certain embodiments to realize the subnets of FIGS. 2A and B. Specifically, each instance of processing network logic 210 communicates with the switch fabric 115 via a PtP links 216 of interconnect 110 . Likewise, the control node has multiple instances of switch logic 214 and each communicates over a PtP connection 216 to the switch fabric. The virtual interfaces of FIG. 2B include the logic to convey information over these physical links, as will be described further below.

To create and configure such networks, an administrator defines the network topology of a PAN and specifies (e.g., via a utility within the management software 135) MAC address assignments of the various nodes. The MAC address is virtual, identifying a virtual interface, and not tied to any specific physical node. Under certain embodiments, MAC addresses follow the IEEE 48 bit address format, but in which the contents include a "locally administered" bit (set to 1), the serial number of the control node 120 on which the virtual interface was originally defined (more below), and a count value from a persistent sequence counter on the control node that is kept in NVRAM in the control node.

These MACs will be used to identify the nodes (as is conventional) at a layer 2 level. For example, in replying to ARP requests (whether from a node internal to the PAN or on an external network) these MACs will be included in the ARP reply.

The control node-side networking logic maintains data structures that contain information reflecting the connectivity of the LAN (e.g., which nodes may communicate to which other nodes). The control node logic also allocates and assigns VI (or RVI) mappings to the defined MAC addresses and allocates and assigns VIs or (RVIs) between the control nodes and between the control nodes and the processing nodes. In the example of FIG. 2A, the logic would allocate and assign VIs 212 of FIG. 2B. (The naming of the VIs and RVIs in some embodiments is a consequence of the switching fabric and the switch fabric manager logic employed.)

As each processor boots, BIOS-based boot logic initializes each processor 106 of the node 105 and, among other things, establishes a (or discovers the) VI 212 to the control node logic. The processor node then obtains from the control node relevant data link information, such as the processor node's MAC address, and the MAC identities of other devices within the same data link configuration. Each processor then registers its IP address with the control node, which then binds the IP address to the node and an RVI (e.g., the RVI on which the registration arrived). In this fashion, the control node will be able to bind IP addresses for each virtual MAC for each node on a subnet. In addition to the above, the processor node also obtains the RVI or VI-related information for its connections to other nodes or to control node networking logic.

Thus, after boot and initialization, the various processor nodes should understand their layer 2, data link connectivity. As will be explained below, layer 3 (IP) connectivity and specifically layer 3 to layer 2 associations are determined during normal processing of the processors as a consequence of the address resolution protocol.

FIG. 3A details the processor-side networking logic 210 and FIG. 3B details the control node-side networking 310 logic of certain embodiments. The processor side logic 210 includes IP stack 305 , virtual network driver 310 , ARP logic 350 , RCLAN layer 315 , and redundant Giganet drivers $320a,b$. The control node-side logic 310 includes redundant Giganet drivers $325a,b$, RCLAN layer 330 , virtual Cluster proxy logic 360 , virtual LAN server 335 , ARP server proxy 355 , virtual LAN proxy 340 , and physical LAN drivers 345 .

IP Stack

The IP stack 305 is the communication protocol stack provided with the operating system (e.g., Linux) used by the processing nodes 106 . The IP stack provides a layer 3 interface for the applications and operating system executing on a processor 106 to communicate with the simulated Ethernet network. The IP stack provides packets of information to the virtual Ethernet layer 310 in conjunction with providing a layer 3, IP address as a destination for that packet. The IP stack logic is conventional except that certain embodiment avoid check sum calculations and logic.

Virtual Ethernet Driver

The virtual Ethernet driver 310 will appear to the IP stack 305 like a "real" Ethernet driver. In this regard, the virtual Ethernet driver 310 receives IP packets or datagrams from the IP stack for subsequent transmission on the network, and

it receives packet information from the network to be delivered to the stack as an IP packet.

The stack builds the MAC header. The "normal" Ethernet code in the stack may be used. The virtual Ethernet driver receives the packet with the MAC header already built and the correct MAC address already in the header.

In material part and with reference to FIGS. 4A-C, the virtual Ethernet driver 310 dequeues 405 outgoing IP datagrams so that the packet may be sent on the network. The standard IP stack ARP logic is used. The driver, as will be explained below, intercepts all ARP packets entering and leaving the system to modify them so that the proper information ends up in each node's ARP tables. The normal ARP logic places the correct MAC address in the link layer header of the outgoing packet before the packet is queued to the Ethernet driver. The driver then just examines the link layer header and destination MAC to determine how to send the packet. The driver does not directly manipulate the ARP table (except for the occasional invalidation of ARP entries).

The driver 310 determines 415 whether ARP logic 350 has MAC address information (more below) associated with the IP address in the dequeued packet. If the ARP logic 350 has the information, the information is used to send 420 the packet accordingly. If the ARP logic 350 does not have the information, the driver needs to determine such information, and in certain preferred embodiments, this information is obtained as a result of an implementation of the ARP protocol as discussed in connection with FIGS. 4B-C.

If the ARP logic 350 has the MAC address information, the driver analyzes the information returned from the ARP logic 350 to determine where and how to send the packet. Specifically, the driver looks at the address to determine whether the MAC address is in a valid format or in a particular invalid format. For example, in one embodiment, internal nodes (i.e., PAN nodes internal to the platform) are signaled through a combination of setting the locally administered bit, the multicast bit, and another predefined bit pattern in the first byte of the MAC address. The overarching pattern is one which is highly improbable of being a valid pattern.

If the MAC address returned from the ARP logic is in a valid format, the IP address associated with that MAC address is for a node external at least to the relevant subnet and in preferred embodiments is external to the platform. To deliver such a packet, the driver prepends the packet with a TLV (type-length-value) header. The logic then sends the packet to the control node over a pre-established VI. The control node then handles the rest of the transmission as appropriate.

If the MAC address information returned from the ARP logic 350 is in an particular invalid format, the invalid format signals that the IP-addressed node is to an internal node, and the information in the MAC address information is used to help identify the VI (or RVI) directly connecting the two processing nodes. For example, the ARP table entry may hold information identifying the RVI 212 to use to send the packet, e.g., 212₁₋₂, to another processing node. The driver prepends the packet with a TLV header. It then places address information into the header as well as information identifying the Ethernet protocol type. The logic then selects the appropriate VI (or RVI) on which to send the encapsulated packet. If that VI (or RVI) is operating satisfactorily it is used to carry the packet; if it is operating unsatisfactorily the packet is sent to the control node switch logic (more below) so that the switch logic can send it to the appropriate node. Though the ARP table may contain information to actually specify the RVI to use, many other techniques may

be employed. For example, the information in the table may indirectly provide such information, e.g., by pointing to the information of interest or otherwise identifying the information of interest though not contain it.

For any multicast or broadcast type messages, the driver sends the message to the control node on a defined VI. The control node then clones the packet and sends it to all nodes (excluding the sending node) and the uplink accordingly.

If there is no ARP mapping then the upper layers would never have sent the packet to the driver. If there is no datalink layer mapping available, the packet is put aside until ARP resolution is completed. Once the ARP layer has finished ARPing, the packets held back pending ARP get their datalink headers build and the packets are then sent to the driver.

If the ARP logic has no mapping for an IP address of an IP packet from the IP stack and, consequently, the driver 310 is unable to determine the associated addressing information (i.e., MAC address or RVI-related information), the driver obtains such information by following the ARP protocol. Referring to FIGS. 4B-C, the driver builds 425 an ARP request packet containing the relevant IP address for which there is no MAC mapping in the local ARP table. The node then prepends 430 the ARP packet with a TLV-type header. The ARP request is then sent via a dedicated RVI to the control node-side networking logic—specifically, the virtual LAN server 335.

As will be discussed in more detail below, the ARP request packet is processed 435 by the control node and broadcast 440 to the relevant nodes. For example, the control node will flag whether the requesting node is part of an IP service cluster.

The Ethernet driver logic 310 at the relevant nodes receives 445 the ARP reply, and determines 450 if it is the target of the ARP request by comparing the target IP address with a list of locally configured IP addresses by making calls to the node's IP stack. If it is not the target, it passes up the packet without modification. If it is the target, the driver creates 460 a local MAC header from the TLV header and updates 465 the local ARP table and creates an ARP reply. The driver modifies the information in the ARP request (mainly the source MAC) and then passes the ARP request up normally for the upper layers to handle. It is the upper layers that form the ARP reply when necessary. The reply among other things contains the MAC address of the replying node and has a bit set in the TLV header indicating that the reply is from a local node. In this regard, the node responds according to IETF-type ARP semantics (in contrast to ATM ARP protocols in which ARP replies are handled centrally). The reply is then sent 470.

As will be explained in more detail below, the control node logic 335 receives 473 the reply and modifies it. For example, the control node may substitute the MAC address of a replying, internal node with information identifying the source cabinet, processing node number, RVI connection number, channel, virtual interface number, and virtual LAN name. Once the ARP reply is modified the control node logic then sends 475 the ARP reply to an appropriate node, i.e., the node that sent the ARP request, or in specific instances to the load balancer in an IP service cluster, discussed below.

Eventually, an encapsulated ARP reply is received 480. If the replying node is an external node, the ARP reply contains the MAC address of the replying node. If the replying node is an internal node, the ARP reply instead contains information identifying the relevant RVI to communicate with the node. In either case, the local table is updated 485.

The pending datagram is dequeued **487**, and the appropriate RVI is selected **493**. As discussed above, the appropriate RVI is selected based on whether the target node is internal or external. A TLV header is prepended to the packet and sent **495**.

For communications within a virtual LAN the maximum transmission unit (MTU) is configured as 16896 bytes. Even though the configured MTU is 16896 bytes, the Ethernet driver **310** recognizes when a packet is being sent to an external network. Through the use of path MTU discovery, ICMP and IP stack changes, the path MTU is changed at the source node **105**. This mechanism is also used to trigger packet check summing.

Certain embodiments of the invention support promiscuous mode through a combination of logic at the virtual LAN server **335** and in the virtual LAN drivers **310**. When a virtual LAN driver **310** receives a promiscuous mode message from the virtual LAN server **335**, the message contains information about the identity of the receiver desiring to enter promiscuous mode. This information includes the receiver's location (cabinet, node, etc), the interface number of the promiscuous virtual interface **310** on the receiver (required for demultiplexing packets), and the name of the virtual LAN to which the receiver belongs. This information is then used by the driver **310** to determine how to send promiscuous packets to the receiver (which RVI or other mechanism to use to send the packets). The virtual interface **310** maintains a list of promiscuous listeners on the same virtual LAN. When a sending node receives a promiscuous mode message it will update its promiscuous list accordingly.

When a packet is transmitted over a virtual Ethernet driver **310**, this list will be examined. If the list is not empty, then the virtual Ethernet interface **310** will do the following:

If the outgoing packet is being broadcast or multicast, no promiscuous copy will be sent. The normal broadcast operation will transmit the packet to the promiscuous listener(s).

If the packet is a unicast packet with a destination other than the promiscuous listener, the packet will be cloned and sent to the promiscuous listeners.

The header TLV includes extra information the destination can use to demultiplex and validate the incoming packet. Part of this information is the destination virtual Ethernet interface number (destination device number on the receiving node). Since these can be different between the actual packet destination and the promiscuous destination, this header cannot simply be cloned. Thus, memory will have to be allocated for each header for each packet clone to each promiscuous listener. When the packet header for a promiscuous packet is built the packet type will be set to indicate that the packet was a promiscuous transmission rather than a unicast transmission.

The virtual Ethernet driver **310** is also responsible for handling the redundant control node connections. For example, the virtual Ethernet drivers will periodically test end-to-end connectivity by sending a heartbeat TLV to each connected RVI. This will allow virtual Ethernet drivers to determine if a node has stopped responding or whether a stopped node has started to respond again. When an RVI or control node **120** is determined to be down, the Ethernet driver will send traffic through the surviving control node. If both control nodes are functional the driver **310** will attempt to load balance traffic between the two nodes.

Certain embodiments of the invention provide performance improvements. For example, with modifications to the IP stack **305**, packets sent only within the platform **100**

are not check summed since all elements of the platform **100** provide error detection and guaranteed data delivery.

In addition, for communications within a PAN (or even within a platform **100**) the RVI may be configured so that the packets may be larger than the maximum size permitted by Ethernet. Thus, while the model emulates Ethernet behavior in certain embodiments maximum packet size may be violated to improve performance. The actual packet size will be negotiated as part of the data link layer.

Failure of a control node is detected either by a notification from the RCLAN layer, or by a failure of heartbeat TLVs. If a control node fails the Ethernet driver **310** will send traffic only to the remaining control node. The Ethernet driver **310** will recognize the recovery of a control node via notification from the RCLAN layer or the resumption of heartbeat TLVs. Once a control node has recovered, the Ethernet driver **310** will resume load balancing. If a node detects that it cannot communicate with another node via a direct RVI (as outlined above) the node attempts to communicate via the control node, acting as a switch. Such failure may be signaled by the lower RCLAN layer, for example from failure to receive a virtual interface acknowledgement or from failures detected through heartbeat mechanisms. In this instance, the driver marks bits in the TLV header accordingly to indicate that the message is to be unicast and sends the packet to the control node so that it can send the packet to the desired node (e.g., based on the IP address, if necessary).

RCLAN Layer

The RCLAN layer **315** is responsible for handling the redundancy, fail-over and load balancing logic of the redundant interconnect NICs **107**. This includes detecting failures, re-routing traffic over a redundant connection on failures, load balancing, and reporting inability to deliver traffic back to the virtual network drivers **310**. The virtual ethernet drivers **310** expect to be notified asynchronously when there is a fatal error on any RVI that makes the RVI unusable or if any RVI is taken down for any reason.

Under normal circumstances the virtual network driver **310** on each processor will attempt to load balance outgoing packets between available control nodes. This can be done via simple round-robin alternation between available control nodes, or by keeping track of how many bytes have been transmitted on each and always transmitting on the control nodes through which fewest bytes have been sent.

The RCLAN provides high bandwidth (224 MB/sec each way) low latency reliable asynchronous point-to-point communication between kernels. The sender of the data is notified if the data cannot be delivered and a best effort will be made to deliver it. The RCLAN uses two Gigaset clan 1000 cards to provide redundant communication paths between kernels. It seamlessly recovers single failures in the clan 1000 cards or the Gigaset switches. It detects lost data and data errors and resends the data if needed. Communication will not be disrupted as long as one of the connections is partially working, e.g., the error rate does not exceed 5%. Clients of the RCLAN include the RPC mechanism, the remote SCSI mechanism, and remote Ethernet. The RCLAN also provide a simple form of flow control. Low latency and high concurrency are achieved by allowing multiple simultaneous requests for each device to be sent by the processor node to the control node, so that they can be forwarded to the device as soon as possible or, alternatively so that they can be queued for completion as close to the device as possible as opposed to queuing all requests on the processor node.

11

The RCLAN layer 330 on the control node-side operates analogously to the above.

Giganet Driver

The Giganet driver logic 320 is the logic responsible for providing an interface to the Giganet NIC 107, whether on a processor 106 or control node 120. In short, the Giganet driver logic establishes VI connections, associated by VI id's, so that the higher layers, e.g., RCLAN 315 and Ethernet driver 310, need only understand the semantics of VI's.

Giganet driver logic 320 is responsible for allocating memory in each node for buffers and queues for the VI's, and for conditioning the NIC 107 to know about the connection and its memory allocation. Certain embodiments use VI connections provided by the Giganet driver. The Giganet NIC driver code establishes a Virtual Interface pair (i.e., VI) and assigns it to a corresponding virtual interface id.

Each VI is a bi-directional connection established between one Giganet port and another, or more precisely between memory buffers and memory queues on one node to buffers and queues on another. The allocation of ports and memory is handled by the NIC drivers as stated above. Data is transmitted by placing it into a buffer the NIC knows about and triggering action by writing to a specific memory-mapped register. On the receiving side, the data appears in a buffer and completion status appears in a queue. The data never need be copied if the sending and receiving programs are capable of producing and consuming messages in the connection's buffers. The transmission can even be direct from application program to application program if the operating system memory-maps the connection's buffers and control registers into application address space. Each Giganet port can support 1024 simultaneous VI connections over it and keep them separate from each other with hardware protection, so the operating system as well as disparate applications can safely share a single port. Under one embodiment of the invention, 14 VI connections may be established simultaneously from every port to every other port.

In preferred embodiments, the NIC drivers establish VI connections in redundant pairs, with one connection of the pair going through one of the two switch fabrics 115a,b and the other through the other switch. Moreover, in preferred embodiments, data is sent alternately on the two legs of the pair, equalizing load on the switches. Alternatively, the redundant pairs may be used in fail-over manner.

All the connection pairs established by the node persist as long as the operating system remains up. Establishment of a connection pair to simulate an Ethernet connection is intended to be analogous to, and as persistent as, physically plugging in a cable between network interface cards. If a node's defined configuration changes while its operating system is running, then applicable redundant Virtual Interface connection pairs will be established or discarded at the time of the change.

The Giganet driver logic 325 on the control node-side operates analogously to the above.

Virtual LAN Server

The virtual LAN server logic 335 facilitates the emulation of an Ethernet network over the underlying NBMA network. The virtual LAN server logic

1. manages membership to a corresponding virtual LAN;
2. provides RVI mapping and management;
3. ARP processing and IP mapping to RVI;

12

4. provides broadcast and multicast services;
5. facilitates bridging and routing to other domains; and
6. manages service clusters.

1. Virtual LAN Membership Management

Administrators configure the virtual LANs using management application 135. Assignment and configuration of IP addresses on virtual LANs may be done in the same way as on an "ordinary" subnet. The choice of IP addresses to use is dependent on the external visibility of nodes on a virtual LAN. If the virtual LAN is not globally visible (either not visible outside the platform 100, or from the Internet), private IP addresses should be used. Otherwise, IP addresses must be configured from the range provided by the internet service provider (ISP) that provides the Internet connectivity. In general, virtual LAN IP address assignment must be treated the same as normal LAN IP address assignment. Configuration files stored on the local disks of the control node 120 define the IP addresses within a virtual LAN. For the purposes of a virtual network interface, an IP alias just creates another IP to RVI mapping on the virtual LAN server logic 335. Each processor may configure multiple virtual interfaces as needed. The primary restrictions on the creation and configuration of virtual network interfaces are IP address allocation and configuration.

Each virtual LAN has a corresponding instance of server logic 335 that executes on both of the control nodes 120 and a number of nodes executing on the processor nodes 105. The topology is defined by the administrator.

Each virtual LAN server 335 is configured to manage exactly one broadcast domain, and any number of layer 3 (IP) subnets may be present on the given layer 2 broadcast domain. The servers 335 are configured and created in response to administrator commands to create virtual LANs.

When a processor 106 boots and configures its virtual networks, it connects to the virtual LAN server 335 via a special management RVI. The processors then obtain their data link configuration information, such as the virtual MAC addresses assigned to it, virtual LAN membership information and the like. The virtual LAN server 335 will determine and confirm that the processor attempting to connect to it is properly a member of the virtual LAN that that server 335 is servicing. If the processor is not a virtual LAN member, the connection to the server is rejected. If it is a member, the virtual network driver 310 registers its IP address with the virtual LAN server. (The IP address is provided by the IP stack 305 when the driver 310 is configured.) The virtual LAN server then binds that IP address to an RVI on which the registration arrived. This enables the virtual LAN server to find the processor associated with a specific IP address. Additionally, the association of IP addresses with a processor can be performed via the virtual LAN management interface 135. The latter method is necessary to properly configure cluster IP addresses or IP addresses with special handling, discussed below.

2. RVI Mapping and Management

As outlined above, certain embodiments use RVIs to connect nodes at the data link layer and to form control connections. Some of these connections are created and assigned as part of control nodes booting and initialization. The data link layer connections are used for the reasons described above. The control connections are used to exchange management, configuration, and health information.

13

Some RVI connections are between nodes for unicast traffic, e.g., 212_{1,2}. Other RVI connections are to the virtual LAN server logic 335 so that the server can handle the requests, e.g., ARP traffic, broadcasts, and so on. To create the RVI the virtual LAN server 335 creates and removes RVIs through calls to a Giganet switch manager 360 (provided with the switch fabric and Giganet NICs). The switch manager may execute on the control nodes 120 and cooperates with the Giganet drivers to create the RVIs.

With regard to processor connections, as nodes register with the virtual LAN server 335, the virtual LAN server creates and assigns virtual MAC addresses for the nodes, as described above. In conjunction with this, the virtual LAN server logic maintains data structures reflecting the topology and MAC assignments for the various nodes. The virtual LAN server logic then creates corresponding RVIs for the unicast paths between nodes. These RVIs are subsequently allocated and made known to the nodes during the nodes booting. Moreover, the RVIs are also associated with IP addresses during the virtual LAN server's handling of ARP traffic. The RVI connections are torn down if a node is removed from the topology.

If a node 106 at one end of an established RVI connection is rebooted, the two operating systems of the each end of the connection, and RVI management logic re-establish the connection. Software using the connection on the processing node that remained up will be unaware that anything happened to the connection itself. Whether or not the software notices or cares that the software at the other end was rebooted depends upon what it is using the connection for and the extent to which the rebooted end is able to re-establish its state from persistent storage. For example, any software communicating via Transmission Control Protocol (TCP) will notice that all TCP sessions are closed by a reboot. On the other hand, Network File System (NFS) access is stateless and not affected by a reboot if it occurs within an allowed timeout period.

Should a node be unable to send a packet on a direct RVI at any time, it can always attempt to send the packet to a destination via the virtual LAN server 335. Since the virtual LAN server 335 is connected to all virtual Ethernet driver 310 interfaces on the virtual LAN via the control connections, virtual LAN server 335 can also serve as the packet relay mechanism of last resort.

With regard to the connections to the virtual LAN server 335, certain embodiments use virtual Ethernet drivers 310 that algorithmically determine the RVI that it ought to use to connect to its associated virtual LAN server 335. The algorithm, depending on the embodiment, may need to consider identification information such as cabinet number to identify the RVI.

3. ARP Processing and IP Mapping to RVIs

As explained above, the virtual Ethernet drivers 310 of certain embodiments support ARP. In these embodiments, ARP processing is used to advantage to create mappings at the nodes between IP addresses and RVIs that may be used to carry unicast traffic, including IP packets, between nodes.

To do this, the virtual Ethernet drivers 310 send ARP packet requests and replies to the virtual LAN server 335 via a dedicated RVI. The virtual LAN server 335, and specifically ARP server logic 355, handles the packets by adding information to the packet header. As was explained above, this information facilitates identification of the source and target and identifies the RVI that may be used between the nodes.

14

The ARP server logic 355 receives the ARP requests, processes the TLV header, and broadcasts the request to all relevant nodes on the internal platform and the external network if appropriate. Among other things, the server logic 355 determines who should receive the ARP reply, resulting from the request. For example, if the source is a clustered IP address, the reply should be sent to the cluster load balancer, not necessarily the source of the ARP request. The server logic 355 indicates such by including information in the TLV header of the ARP request, so that the target of the ARP replies accordingly. The server 335 will process the ARP packet by including further information in the appended header and broadcast the packet to the nodes in the relevant domain. For example, the modified header may include information identifying the source cabinet, processing node number, RVI connection number, channel, virtual interface number, and virtual LAN name (some of which is only known by the server 335).

The ARP replies are received by the server logic 355, which then maps the MAC information in the reply to corresponding RVI related information. The RVI-related information is placed in the target MAC entry of the reply and sent to the appropriate source node (e.g., may be the sender of the request, but in some instances such as with clustered IP addresses may be a different node).

4. Broadcast and Multicast Services

As outlined above, broadcasts are handled by receiving the packet on a dedicated RVI. The packet is then cloned by the server 335 and unicast to all virtual interfaces 310 in the relevant broadcast domain.

The same approach may be used for multicast. All multicast packets will be reflected off the virtual LAN server. Under some alternative embodiments, the virtual LAN server will treat multicast the same as broadcast and rely on IP filtering on each node to filter out unwanted packets.

When an application wishes to send or receive multicast addresses it must first join a multicast group. When a process on a processor performs a multicast join, the processor virtual network driver 310 sends a join request to the virtual LAN server 335 via a dedicated RVI. The virtual LAN server then configures a specific multicast MAC address on the interface and informs the LAN Proxy 340, discussed below, as necessary. The Proxy 340 will have to keep track of use counts on specific multicast groups so a multicast address is only removed when no processor belongs to that multicast group.

5. Bridging and Routing to other Domains

From the perspective of system 100, the external network 125 may operate in one of two modes: filtered or unfiltered. In filtered mode a single MAC address for the entire system is used for all outgoing packets. This hides the virtual MAC addresses of a processing node 107 behind the Virtual LAN Proxy 340 and makes the system appear as a single node on the network 125 (or as multiple nodes behind a bridge or proxy). Because this doesn't expose unique link layer information for each internal node 107 some other unique identifier is required to properly deliver incoming packets. When running in filter mode, the destination IP address of each incoming packet is used to uniquely identify the intended recipient since the MAC address will only identify the system. In unfiltered mode the virtual MACs of a node 107 are visible outside the system so that they may be used to

15

direct incoming traffic. That is, filtered mode mandates layer 3 switching while unfiltered mode allows layer 2 switching. Filtered mode requires that some component (in this case the Virtual LAN Proxy 340) perform replacement of node virtual MAC addresses with the MAC address of the external network 125 on all outgoing packets.

Some embodiments support the ability for a virtual LAN to be connected to external networks. Consequently, the virtual LAN will have to handle IP addresses not configured locally. To address this, one embodiment imposes a limit that each virtual LAN so connected be restricted to one external broadcast domain. IP addresses and subnet assignments for the internal nodes of the virtual LAN will have to be in accordance with the external domain.

The virtual LAN server 335 services the external connection by effectively acting as a data link layer bridge in that it moves packets between the external Ethernet driver 345 and internal processors and performs no IP processing. However, unlike like a data link layer bridge, the server cannot always rely on distinctive layer two addresses from the external network to internal nodes and instead the connection may use layer 3 (IP) information to make the bridging decisions. To do this, the external connection software extracts IP address information from incoming packets and it uses this information to identify the correct node 106 so that it may move the packet to that node.

A virtual LAN server 335 having an attached external broadcast domain has to intercept and process packets from and to the external domain so that external nodes have a consistent view of the subnet(s) in the broadcast domain.

When virtual LAN server 335 having an attached external broadcast domain receives an ARP request from an external node it will relay the request to all internal nodes. The correct node will then compose the reply and send the reply back to the requester through the virtual LAN server 335. The virtual LAN server cooperates with the virtual LAN Proxy 340 so that the Proxy may handle any necessary MAC address translation on outgoing requests. All ARP Replies and ARP advertisements from external sources will be relayed directly to the target nodes.

Virtual Ethernet interfaces 310 will send all unicast packets with an external destination to the virtual LAN server 335 over the control connection RVI. (External destinations may be recognized by the driver by the MAC address format.) The virtual LAN server will then move the packet to the external network 125 accordingly.

If the virtual LAN server 335 receives a broadcast or multicast packet from an internal node it relays the packet to the external network in addition to relaying the packet to all internal virtual LAN members. If the virtual LAN server 335 receives a broadcast or multicast packet from an external source it relays the packet to all attached internal nodes.

Under certain embodiments, interconnecting virtual LANs through the use of IP routers or firewalls is accomplished using analogous mechanisms to those used in interconnecting physical LANs. One processor is configured on both LANs, and the Linux kernel on that processor must have routing (and possibly IP masquerading) enabled. Normal IP subnetting and routing semantics will always be maintained, even for two nodes located in the same platform.

A processor could be configured as a router between two external subnets, between and external and internal subnet, and between two internal subnets. When an internal node is sending a packet through a router there are no problems because of the point-to-point topology of the internal network. The sender will send directly to the router (i.e.,

16

processor so configured with routing logic) without the intervention of the virtual LAN server (i.e., typical processor to processor communication, discussed above).

When an external node sends a packet to an internal router, and the external network 125 is running in filtered mode, the destination MAC address of the incoming packet will be that of the platform 100. Thus the MAC address can not be used to uniquely identify the packet destination node. For a packet whose destination is an internal node on the virtual LAN, the destination IP address in the IP header is used to direct the packet to the proper destination node. However, because routers are not final destinations, the destination IP address in the IP header is that of the final destination rather than that of the next hop (which is the internal router). Thus, there is nothing in the incoming packet that can be used to direct it to the correct internal node. To handle this situation, one embodiment imposes a limit of no more than one router exposed to an external network on a virtual LAN. This router is registered with the virtual LAN server 335 as a default destination so that incoming packets with no valid destination will be directed to this default node.

When an external node sends a packet to an internal router and the external network 125 is running in unfiltered mode, the destination MAC address of the incoming packet will be the virtual MAC address of the internal destination node. The LAN Server 335 will then use this virtual MAC to send the packet directly to the destination internal node. In this case any number of internal nodes may be functioning as routers as the incoming packet's MAC address will uniquely identify the destination node.

If a configuration requires multiple routers on a subnet, one router can be picked as the exposed router. This router in turn could route to the other routers as necessary.

Under certain embodiments, router redundancy is provided, by making a router a clustered service and load balancing or failing over on a stateless basis (i.e., every IP packet rather than per-TCP connection).

Certain embodiments of the invention support promiscuous mode functionality by providing switch semantics in which a given port may be designated as a promiscuous port so that all traffic passing through the switch is repeated on the promiscuous port. The nodes that are allowed to listen in promiscuous mode will be assigned administratively at the virtual LAN server.

When a virtual Ethernet interface 310 enters promiscuous receive mode it will send a message to the virtual LAN server 335 over the management RVI. This message will contain all the information about the virtual Ethernet interface 310 entering promiscuous mode. When the virtual LAN Server receives a promiscuous mode message from a node, it will check its configuration information to determine if the node is allowed to listen promiscuously. If not, the virtual LAN Server will drop the promiscuous mode message without further processing. If the node is allowed to enter promiscuous mode, the virtual LAN server will broadcast the promiscuous mode message to all other nodes on the virtual LAN. The virtual LAN server will also mark the node as being promiscuous so that it can forward copies of incoming external packets to it. When a promiscuous listener detects any change in its RVI configuration it will send a promiscuous mode message to the virtual LAN to update the state of all other nodes on the relevant broadcast domain. This will update any nodes entering or leaving a virtual LAN. When a virtual Ethernet interface 310 leaves promiscuous it will send the virtual LAN server a message informing it that the interface is leaving promiscuous mode. The

virtual LAN server will then send this message to all other nodes on the virtual LAN. Promiscuous settings will allow for placing an external connection in promiscuous mode when any internal virtual interface is a promiscuous listener. This will make the traffic external to the platform (but on the same virtual LAN) available to the promiscuous listener.

6. Managing Service Clusters

A service cluster is a set of services available at one or more IP address (or host names). Examples of these services are HTTP, FTP, telnet, NFS, etc. An IP address and port number pair represents a specific service type (though not a service instance) offered by the cluster to clients, including clients on the external network 125.

FIG. 5 shows how certain embodiments present a virtual cluster 405 of services as a single virtual host to the Internet or other external network 125 via a cluster IP address. All the services of the cluster 505 are addressed through a single IP address, through different ports at that IP address. In the example of FIG. 5, service B is a load balanced service.

With reference to FIG. 3B, virtual clusters are supported by the inclusion of virtual cluster proxy (VCP) logic 360 which cooperates with the virtual LAN server 335. In short, VCP 360 is responsible for handling distribution of incoming connections, port filters, and real server connections for each configured virtual IP address. There will be one VCP for each clustered IP address configured.

When a packet arrives on the virtual cluster IP address, the virtual LAN Proxy logic 340 will send the packet to the VCP 360 for processing. The VCP will then decide where to send the packet based on the packet contents, its internal connection state cache, any load balancing algorithms being applied to incoming traffic, and the availability of configured services. The VCP will relay incoming packets based on both the destination IP address as well as the TCP or UDP port number. Further, it will only distribute packets destined for port numbers known to the VCP (or for existing TCP connections). It is the configuration of these ports, and the mapping of the port number to one or more processors that creates the virtual cluster and makes specific service instances available in the cluster. If multiple instances of the same service from multiple application processors are configured then the VCP can load balance between the service instances.

The VCP 360 maintains a cache of all active connections that exist on the cluster's IP address. Any load balancing decisions that are made will only be made when a new connection is established between the client and a service. Once the connection has been set up, the VCP will use the source and destination information in the incoming packet header to make sure all packets in a TCP stream get routed to the same processor 106 configured to provide the service. In the absence of the ability to determine a client session (for example, HTTP sessions), the actual connection/load balancing mapping cache will route packets based on client address so that subsequent connections from the same client goes to the same processor (making a client session persistent or "sticky"). Session persistence should be selectable on a service port number basis since only certain types of services require session persistence.

Replies to ARP requests, and routing of ARP replies, is handled by the VCP. When a processor sends any ARP packet, it will send it out through the Virtual Ethernet driver 310. The packet will then be sent to the virtual LAN Server 335 for normal ARP processing. The virtual LAN server will broadcast the packet as usual, but will make sure it doesn't

get broadcast to any member of the cluster (not just the sender). It will also place information in the packet header TLV that indicates to the ARP target that the ARP source can only be reached through the virtual LAN server and specifically through the load balancer. The ARP target, whether internal or external, will process the ARP request normally and send a reply back through the virtual LAN server. Because the source of the ARP was a cluster IP address, the virtual LAN server will be unable to determine which processor sent out the original request. Thus, the virtual LAN Server will send the reply to each cluster member so that they can handle it properly. When an ARP packet is sent by a source with a cluster IP address as the target, the virtual LAN server will send the request to every cluster member. Each cluster member will receive the ARP request and process it normally. They will then compose an ARP reply and send it back to the source via the virtual LAN server. When the virtual LAN server receives any ARP reply from a cluster member it will drop that reply, but the virtual LAN server will compose and send an ARP reply to the ARP source. Thus, the virtual LAN Server will respond to all ARPs of the cluster IP address. The ARP reply will contain the information necessary for the ARP source to send all packets for the cluster IP address to the VCP. For external ARP sources, this will simply be an ARP reply with the external MAC address as the source hardware address. For internal ARP sources this will be the information necessary to tell the source to send packets for the cluster IP address down the virtual LAN management RVI rather than through a directly connected RVI. Any gratuitous ARP packets that are received will be forwarded to all cluster members. Any gratuitous ARP packets sent by a cluster member will be sent normally.

Virtual LAN Proxy

The virtual LAN Proxy 340 performs the basic coordination of the physical network resources among all the processors that have virtual interfaces to the external physical network 125. It bridges virtual LAN server 335 to the external network 125. When the external network 125 is running in filtered mode the Virtual LAN Proxy 340 will convert the internal virtual MAC addresses from each node to the single external MAC assigned to the system 100. When the external network 125 is operating in unfiltered mode no such MAC translation is required. The Virtual LAN Proxy 340 also performs insertion and removal of IEEE 802.1Q Virtual LAN ID tagging information, and demultiplexing packets based on their VLAN Ids. It also serializes access to the physical Ethernet interface 129 and co-ordinates the allocation and removal of MAC addresses, such as multicast addresses, on the physical network.

When the external network 125 is running in filtered mode and the virtual LAN Proxy 340 receives outgoing packets (ARP or otherwise) from a virtual LAN server 335, it replace the internal format MAC address with the MAC address of the physical Ethernet device 129 as the source MAC address. When the External Network 125 is running in unfiltered mode no such replacement is required.

When the virtual LAN Proxy 340 receives incoming ARP packets, it moves the packet to the virtual LAN server 335 which handles the packet and relays the packet on to the correct destination(s). If the ARP packet is a broadcast packet then the packet is relayed to all internal nodes on the virtual LAN. If the packet is a unicast packet the packet is sent only to the destination node. The destination node is determined by the IP address in the ARP packet when the

External Network 125 is running in filtered mode, or by the MAC address in the Ethernet header of the ARP packet (not the MAC address is the ARP packet).

Physical LAN Driver

Under certain embodiments, the connection to the external network 125 is via Gigabit or 100/10baseT Ethernet links connected to the control node. Physical LAN drivers 345 are responsible for interfacing with such links. Packets being sent on the interface will be queued to the device in the normal manner, including placing the packets in socket buffers. The queue used to queue the packets is the one used by the protocol stack to queue packets to the device's transmit routine. For incoming packets, the socket buffer containing the packets will be passed around and the packet data will never be copied (though it will be cloned if needed for multicast operations). Under these embodiments, generic Linux network device drivers may be used in the control node without modification. This facilitates the addition of new devices to the platform without requiring additional device driver work.

The physical network interface 345 is in communication only with the virtual LAN proxy 340. This prevents the control node from using the external connection in any way that would interfere with the operation of the virtual LANs and improves security and isolation of user data, i.e., an administrator may not "sniff" any user's packets.

Load Balancing and Failover

Under some embodiments, the redundant connections to the external network 125 will be used alternately to load balance packet transmission between two redundant interfaces to the external network 125. Other embodiments load balance by configuring each virtual network interface on alternating control nodes so the virtual interfaces are evenly distributed between the two control nodes. Another embodiment transmits through one control node and receives through another.

When in filtered mode, there will be one externally visible MAC address to which external nodes transmit packets for a set of virtual network interfaces. If that adapter goes down, then not only do the virtual network interfaces have to fail over to the other control node, but the MAC address must fail over too so that external nodes can continue to send packets to the MAC address already in the ARP caches. Under one embodiment of the invention, when a failed control node recovers, a single MAC address is manipulated and the MAC address does not have to be remapped on recovery.

Under another embodiment of the invention, load balancing is performed by allowing transmission on both control nodes but only reception through one. The failover case is both send and receive through the same control node. The recovery case is transmission through the recovered control node since that doesn't require any MAC manipulation.

The control node doing reception has IP information for filtering and multicast address information for multicast MAC configuration. This information is needed to process incoming packets and should be failed over should the receiving control node fail. If the transmitting control node fails, virtual network drivers need only start sending outgoing packets only to the receiving control node. No special failover processing is required other than the recognition that the transmitting control node has failed. If the failed control node recovers the virtual network drivers can resume

sending outgoing packets to the recovered control nodes without any additional special recovery processing. If the receiving control node fails then the transmitting control node must assume the receiving interface role. To do this, it must configure all MAC addresses on its physical interface to enable packet reception. Alternately, both control nodes could have the same MAC address configured on their interfaces, but receives could be physically disabled on the Ethernet device by the device driver until an control node is ready to receive packets. Then failover would simply enable receives on the device.

Because the interfaces must be configured with multicast MAC addresses when any processor has joined a multicast group, multicast information must be shared between control nodes so that failover will be transparent to the processor. Since the virtual network drivers will have to keep track of multicast group membership anyway, this information will always be available to a LAN Proxy via the virtual LAN server when needed. Thus, a receive failover will result in multicast group membership being queried from virtual network drivers to rebuild the local multicast group membership tables. This operations is low overhead and requires no special processing except during failover and recovery, and doesn't require any special replication of data between control nodes. When receive has failed over and the failed control node recovers, only transmissions will be moved over to the recovered control node. Thus, the algorithm for recovery on virtual network interfaces is to always move transmissions to the recovered control node and leave receive processing where it is.

Virtual service clusters may also use load balancing and failover.

Multicabinet Platforms

Some embodiments allow cabinets to be connected together to form larger platforms. Each cabinet will have at least one control node which will be used for inter-cabinet connections. Each control node will include a virtual LAN server 335 to handle local connections and traffic. One of the servers is configured to be a master, such as the one located on the control node with the external connection for the virtual LAN. The other virtual LAN server will act as proxy servers, or slaves, so that the local processors of those cabinets can participate. The master maintains all virtual LAN state and control while the proxies relay packets between the processors and masters.

Each virtual LAN server proxy maintains a RVI to each master virtual LAN Server. Each local processor will connect to the virtual LAN Server Proxy server just as if it were a master. When a processor connects and registers an IP and MAC address, the proxy will register that IP and MAC address with the master. This will cause the master to bind the addresses to the RVI from the proxy. Thus, the master will contain RVI bindings for all internal nodes, but proxies will contain bindings only for nodes in the same cabinet.

When an processor anywhere in a multicabinet virtual LAN sends any packet to its virtual LAN server, the packet will be relayed to the master for processing. The master will then do normal processing on the packet. The master will relay packets to the proxies as necessary for multicast and broadcast. The master will also relay unicast packets based on the destination IP address of the unicast packet and registered IP addresses on the proxies. Note that on the master, a proxy connection looks very much like a node with many configured IP addresses.

During times when there is no operating system running on a processing node, such as booting or kernel debugging, the node's serial console traffic and boot image requests are routed by switch driver code located in the processing node's kernel debugging software or BIOS to management software running on a control node (not shown). From there, the console traffic can again be accessed either from the high-speed external network 125 or through the control node's management ports. The boot image requests can be satisfied from either the control node's local disks or from partitions out on the external SAN 130. The control node 120 is preferably booted and running normally before anything can be done to an processing node. The control node is itself booted or debugged from its management ports.

Some customers may wish to restrict booting and debugging of controllers to local access only, by plugging their management ports into an on-site computer when needed. Others may choose to allow remote booting and debugging by establishing a secure network segment for management purposes, suitably isolated from the Internet, into which to plug their management ports. Once a controller is booted and running normally, all other management functions for it and for the rest of the platform can be accessed from the high-speed external network 125 as well as the management ports, if permitted by the administrator.

Serial console traffic to and from each processing node 105 is sent by an operating system kernel driver over the switch fabric 115 to management software running on a control node 120. From there, any node's console traffic can be accessed either from the normal, high-speed external network 125 or through either of the control node's management ports.

Storage Architecture

Certain embodiments follow a SCSI model of storage. Each virtual PAN has its own virtualized I/O space and issues SCSI commands and status within such space. Logic at the control node translates or transforms the addresses and commands as necessary from a PAN and transmits them accordingly to the SAN 130 which services the commands. From the perspective of the SAN, the client is the platform 100 and the actual PANs that issued the commands are hidden and anonymous. Because the SAN address space is virtualized, one PAN operating on the platform 100 may have device numbering starting with a device number 1, and a second PAN may also have a device number 1. Yet each of the device number is will correspond to a different, unique portion of SAN storage.

Under preferred embodiments, an administrator can build virtual storage. Each of the PANs will have its own independent perspective of mass storage. Thus, as will be explained below, a first PAN may have a given device/LUN address map to a first location in the SAN, and a second PAN may have the same given device/LUN map to a second, different location in the SAN. Each processor maps a device/LUN address into a major and minor device number, to identify a disk and a partition, for example. Though the major and minor device numbers are perceived as a physical address by the PAN and the processors within a PAN, in effect they are treated by the platform as a virtual address to the mass storage provided by the SAN. That is, the major and minor device numbers of each processor are mapped to corresponding SAN locations.

FIG. 6 illustrates the software components used to implement the storage architecture of certain embodiments. A

configuration component 605, typically executed on a control node 120, is in communication with external SAN 130. A management interface component 610 provides an interface to the configuration component 605 and is in communication with IP network 125 and thus with remote management logic 135 (see FIG. 1). Each processor 106 in the system 100 includes an instance of processor-side storage logic 620. Each such instance 620 communicates via 2 RVI connections 625 to a corresponding instance of control node-side storage logic 615.

In short, the configuration component 605 and interface 610 are responsible for discovering those portions of SAN storage that are allocated to the platform 100 and for allowing an administrator to suballocate portions to specific PANs or processors 106. Storage configuration logic 605 is also responsible for communicating the SAN storage allocations to control node-side logic 615. The processor-side storage logic 620 is responsible for communicating the processor's storage requests over the internal interconnect 110 and storage fabric 115 via dedicated RVIs 625 to the control node-side logic 615. The requests will contain, under certain embodiments, virtual storage addresses and SCSI commands. The control node-side logic is responsible for receiving and handling such commands by identifying the corresponding actual address for the SAN and converting the commands and protocol to the appropriate form for the SAN, for example, including but not limited to, fibre channel (Gigabit Ethernet with iSCSI is another exemplary connectivity).

Configuration Component

The configuration component 605 determines which elements in the SAN 130 are visible to each individual processor 106. It provides a mapping function that translates the device numbers (e.g., SCSI target and LUN) that the processor uses into the device numbers visible to the control nodes through their attached SCSI and Fibre Channel I/O interfaces 128. It also provides an access control function, which prevents processors from accessing external storage devices which are attached to the control nodes but not included in the processors' configuration. The model that is presented to the processor (and to the system administrator and applications/users on that processor) makes it appear as if each processor has its own mass storage devices attached to interfaces on the processor.

Among other things, this functionality allows the software on a processor 106 to be moved to another processor easily. For example, in certain embodiments, the control node via software (without any physical re-cabling) may change the PAN configurations to allow a new processor to access the required devices. Thus, a new processor may be made to inherit the storage personality of another.

Under certain embodiments, the control nodes appear as hosts on the SANs, though alternative embodiments allow the processors to act as such.

As outlined above, the configuration logic discovers the SAN storage allocated to the platform 100 (for example, during platform boot) and this pool is subsequently allocated by an administrator. If discovery is activated later, the control node that performs the discovery operation compares the new view with the prior view. Newly available storage is added to the pool of storage that may be allocated by an administrator. Partitions that disappear that were not assigned are removed from the available pool of storage that may be allocated to PANs. Partitions that disappear that were assigned trigger error messages.

The configuration component 605 allows management software to access and update the information which describes the device mapping between the devices visible to the control nodes 120 and the virtual devices visible to the individual processors 106. It also allows access to control information. The assignments may be identified by the processing node in conjunction with an identification of the simulated SCSI disks, e.g., by name of the simulated controller, cable, unit, or logical unit number (LUN).

Under certain embodiments the interface component 610 cooperates with the configuration component to gather and monitor information and statistics, such as:

- Total number of I/O operations performed
- Total number of bytes transferred
- Total number of read operations performed
- Total number of write operations performed
- Total amount of time I/O was in progress

Processor-side Storage Logic

The processor-side logic 620 of the protocol is implemented as a host adapter module that emulates a SCSI subsystem by providing a low-level virtual interface to in the operating system on the processors 106. The processors 106 use this virtual interface to send SCSI I/O commands to the control nodes 120 for processing.

Under embodiments employing redundant control nodes 120, each processing node 105 will include one instance of logic 620 per control node 120. Under certain embodiments, the processors refer to storage using physical device numbering, rather than logical. That is, the address is specified as a device name to identify the LUN, the SCSI target, channel, host adapter, and control node 120 (e.g., node 120a or 120b). As shown in FIG. 8, one embodiment maps the target (T) and LUN (L) to a host adapter (H), channel (C), mapped target (mT), and mapped LUN (mL).

FIG. 7 shows an exemplary architecture for processor side logic 720. Logic 720 includes a device-type-specific driver (e.g., a disk driver) 705, a mid-level SCSI I/O driver 710, and wrapper and interconnect logic 715.

The device-type-specific driver 705 is a conventional driver provided with the operating system and associated with specific device types.

The mid-level SCSI I/O driver 710 is a conventional mid-level driver that is called by the device-type-specific driver 705 once the driver 705 determines that the device is a SCSI device.

The wrapper and interconnect logic 715 is called by the mid-level SCSI I/O driver 710. This logic provides the SCSI subsystem interface and thus emulates the SCSI subsystem. In certain embodiments that use the Giganet fabric, logic 715 is responsible for wrapping the SCSI commands as necessary and for interacting with the Giganet and RCLAN interface to cause the NIC to send the packets to the control nodes via the dedicated RVIs to the control nodes, described above. The header information for the Giganet packet is modified to indicate that this is a storage packet and includes other information, described below in context. Though not shown in FIG. 7, wrapper logic 715 may use the RCLAN layer to support and utilize redundant interconnects 110 and fabrics 115.

For embodiments that use Giganet fabric 115, the RVIs of connection 725 are assigned virtual interface (VI) numbers from the range of 1024 available VIs. For the two endpoints to communicate, the switch 115 is programmed with a

bi-directional path between the pair (control node switch port, control node VI number), (processor node 105 switch port, processor node VI number).

A separate RVI is used for each type of message sent in either direction. Thus, there is always a receive buffer pending on each RVI for a message that can be sent from the other side of the protocol. In addition, since only one type of message is sent in either direction on each RVI, the receive buffers posted to each of the RVI channels can be sized appropriately for the maximum message length that the protocol will use for that type of message. Under other embodiments, all of the possible message types are multiplexed onto a single RVI, rather than using 2 VIs. The protocol and the message format do not specifically require the use of 2 RVIs, and the messages themselves have message type information in their header so that they could be demultiplexed.

One of the two channels is used to exchange SCSI commands (CMD) and status (STAT) messages. The other channel is used to exchange buffer (BUF) and transmit (TRAN) messages. This channel is also used to handle data payloads of SCSI commands.

CMD messages contain control information, the SCSI command to be performed, and the virtual addresses and sizes of I/O buffers in the node 105. STAT messages contain control information and a completion status code reflecting any errors that may have occurred while processing the SCSI command. BUF messages contain control information and the virtual addresses and sizes of I/O buffers in the control node 120. TRAN messages contain control information and are used to confirm successful transmission of data from node 105 to the control node 120.

The processor side wrapper logic 715 examines the SCSI command to be sent to determine if the command requires the transfer of data and, if so, in what direction. Depending on the analysis, the wrapper logic 715 sets appropriate flag information in the message header accordingly. The section describing the control node-side logic describes how the flag information is utilized.

Under certain embodiments of the invention, the link 725 between processor-side storage logic 720 and control node-side storage logic 715 may be used to convey control messages, not part of the SCSI protocol and not to be communicated to the SAN 130. Instead, these control messages are to be handled by the control node-side logic 715.

The protocol control messages are always generated by the processor-side of the protocol and sent to the control node-side of the protocol over one of two virtual interfaces (VIs) connecting the processor-side logic 720 to the control node-side storage logic 715. The message header used for protocol control operations is the same as a command message header, except that different flag bits are used to distinguish the message as a protocol control message. The control node 120 performs the requested operation and responds over the RVI with a message header that is the same as is used by a status message. In this fashion, a separate RVI for the infrequently used protocol control operations is not needed.

Under certain embodiments using redundant control nodes, the processor-side logic 720 detects certain errors from issued commands and in response re-issues the command to the other control node. This retry may be implemented in a mid-level driver 710.

Under certain embodiments, the control node-side storage logic 715 is implemented as a device driver module. The logic 715 provides a device-level interface to the operating system on the control nodes 120. This device-level interface is also used to access the configuration component 705. When this device driver module is initialized, it responds to protocol messages from all of the processors 106 in the platform 100. All of the configuration activity is introduced through the device-level interface. All of the I/O activity is introduced through messages that are sent and received through the interconnect 110 and switch fabric 115. On the control node 120, there will be one instance of logic 715 per processor node 105 (though it is only shown as one box in FIG. 7). Under certain embodiments, the control node-side logic 715 communicates with the SAN 130 via FCP or FCP-2 protocols, or iSCSI or other protocols that use the SCSI-2 or SCSI-3 command set over various media.

As described above, the processor-side logic sets flags in the RVI message headers indicating whether data flow is associated with the command and, if so, in which direction. The control node-side storage logic 715 receives messages from the processor-side logic and then analyzes the header information to determine how to act, e.g., to allocate buffers or the like. In addition, the logic translates the address information contained in the messages from the processor to the corresponding, mapped SAN address and issues the commands (e.g., via FCP or FCP-2) to the SAN 130.

A SCSI command such as a TEST UNIT READY command, which does not require a SCSI data transfer phase, is handled by the processor-side logic 720 sending a single command on the RVI used for command messages, and by the control node-side logic sending a single status message back over the same RVI. More specifically, the processor-side of the protocol constructs the message with a standard message header, a new sequence number for this command, the desired SCSI target and LUN, the SCSI command to be executed, and a list size of zero. The control node-side of the logic receives the message, extracts the SCSI command information and conveys it to the SAN 130 via interface 128. After the control node has received the command completion callback, it constructs a status message to the processor using a standard message header, the sequence number for this command, the status of the completed command, and optionally the request sense data if the command completed with a check condition status.

A SCSI command such as a READ command, which requires a SCSI data transfer phase to transfer data from the SCSI device into the host memory, is handled by the processor-side logic sending a command message to the control node-side logic 715, and the control node responding with one or more RDMA WRITE operation into memory in the processor node 105, and a single status message from the control node-side logic. More specifically, the processor-side logic 720 constructs a command message with a standard message header, a new sequence number for this command, the desired SCSI target and LUN, the SCSI command to be executed, and a list of regions of memory where the data from the command is to be stored. The control node-side logic 715 allocates temporary memory buffers to store the data from the SCSI operation while the SCSI command is executing on the control node. After the control node-side logic 715 has sent the SCSI command to the SAN 130 for processing and the command has completed it sends the data back to the processor 105 memory with a sequence of one or more RDMA WRITE operations.

It then constructs a status message with a standard message header, the sequence number for this command, the status of the completed command, and optionally the REQUEST SENSE data if the command completed with a SCSI CHECK CONDITION status.

A SCSI command such as a WRITE command, which requires a SCSI data transfer phase to transfer data from the host memory to the SCSI device, is handled by the processor-side logic 720 sending a single command message to the control node-side logic 715, one or more BUF messages from the control node-side logic 715 to the processor-side logic, one or more RDMA WRITE operations from the processor-side storage logic into memory in the control node, one or more TRAN messages from the processor-side logic to the control node-side logic, and a single status message from the control node-side logic back to the processor-side logic. The use of the BUF messages to communicate the location of temporary buffer memory in the control node to the processor-side storage logic and the use of TRAN messages to indicate completion of the RDMA WRITE data transfer is due to the lack of RDMA READ capability in the underlying Giganet fabric. If the underlying fabric supports RDMA READ operations, a different sequence of corresponding actions may be employed. More specifically, the processor-side logic 720 constructs a CMD message with a standard message header, a new sequence number for this command, the desired SCSI target and LUN, and the SCSI command to be executed. The control node-side logic 715 allocates temporary memory buffers to store the data from the SCSI operation while the SCSI command is executing on the control node. The control node-side of the protocol then constructs a BUF message with a standard message header, the sequence number for this command, and a list of regions of virtual memory which are used for the temporary memory buffers on the control node. The processor-side logic 720 then sends the data over to the control node memory with a sequence of one or more RDMA WRITE operations. It then constructs a TRAN message with a standard message header, and the sequence number for this command. After the control node-side logic has sent the SCSI command to the SAN 130 for processing and has received the command completion, it constructs a STAT message with a standard message header, the sequence number for this command, the status of the completed command, and optionally the REQUEST SENSE data if the command completed with a CHECK CONDITION status.

Under some embodiments, the CMD message contains a list of regions of virtual memory from where the data for the command is stored. The BUF and TRAN messages also contain an index field, which allows the control node-side of the protocol to send a separate BUF message for each entry in the region list in the CMD message. The processor-side of the protocol would respond to such a message by performing RDMA WRITE operations for the amount of data described in the BUF message, followed by a TRAN message to indicate the completion of a single segment of data transfer.

The protocol between the processor-side logic 720 and the control node-side logic 715 allows for scatter-gather I/O operations. This functionality allows the data involved in an I/O request to be read from or written to several distinct regions of virtual and/or physical memory. This allows multiple, non-contiguous buffers to be used for the request on the control node.

As stated above, the configuration logic 705 is responsible for discovering the SAN storage allocated to the platform and for interacting with the interface logic 710 so that an administrator may suballocate the storage to specific PANs.

As part of this allocation, the configuration component 705 creates and maintains a storage data structure 915 that includes information identifying the correspondence between processor addresses and actual SAN addresses. FIG. 7 shows such a structure. The correspondence, as described above, may be between the processing node and the identification of the simulated SCSI disks, e.g., by name of the simulated controller, cable, unit, or logical unit number (LUN).

Management Logic

Management logic 135 is used to interface to control node software to provision the PANs. Among other things, the logic 135 allows an administrator to establish the virtual network topology of a PAN, its visibility to the external network (e.g., as a service cluster), and to establish the types of devices on the PAN, e.g., bridges and routing.

The logic 135 also interfaces with the storage management interface logic 710 so that an administrator may define the storage for a PAN during initial allocation or subsequently. The configuration definition includes the storage correspondence (SCSI to SAN) discussed above and access control permissions.

As described above, each of the PANs and each of the processors will have a personality defined by its virtual networking (including a virtual MAC address) and virtual storage. The structures that record such personality may be accessed by management logic, as described below, to implement processor clustering. In addition, they may be accessed by an administrator as described above or with an agent administrator. An agent for example may be used to re-configure a PAN in response to certain events, such as time of day or year, or in response to certain loads on the system.

The operating system software at a processor includes serial console driver code to route console I/O traffic for the node over the Gigaset switch 115 to management software running on a control node. From there, the management software can make any node's console I/O stream accessible via the control node's management ports (its low-speed Ethernet port and its Emergency Management Port) or via the high-speed external network 125, as permitted by an administrator. Console traffic can be logged for audit and history purposes.

Cluster Management Logic

FIG. 9 illustrates the cluster management logic of certain embodiments. The cluster management logic 905 accesses the data structures 910 that record the networking information described above, such as the network topologies of PANs, the MAC address assignments within a PAN and so on. In addition, the cluster management logic 905 accesses the data structures 915 that record the storage correspondence of the various processors 106. Moreover, the cluster management logic 905 accesses a data structure 920 that records free resources such as unallocated processors within the platform 100.

In response to processor error events or administrator commands, the cluster management logic 905 can change the data structures to cause the storage and networking personalities of a given processor to "migrate" to a new processor. In this fashion, the new processor "inherits" the personality of the former processor. The cluster management logic 905 may be caused to do this to swap a new processor in to a PAN to replace a failing one.

The new processor will inherit the MAC address of a former processor and act like the former. The control node will communicate the connectivity information when the

new processor boots, and will update the connectivity information for the non-failing processors as needed. For example, in certain embodiments, the RVI connections for the other processors are updated transparently; that is, the software on the other processors does not need to be involved in establishing connectivity to the newly swapped in processor. Moreover, the new processor will inherit the storage correspondence of the former and consequently inherit the persisted state of the former processor.

Among other advantages this allows a free pool of resources, including processors, to be shared across the entire platform rather than across given PANs. In this way, the free resources (which may be kept as such to improve reliability and fault tolerance of the system) may be used more efficiently.

When a new processor is "swapped in" it will need to re-ARP to learn IP address to MAC address associations.

Alternatives

As each Gigaset port of the switch fabric 115 can support 1024 simultaneous Virtual Interface connections over it and keep them separate from each other with hardware protection, the operating system can safely share a node's Gigaset ports with application programs. This would allow direct connection between application programs without the need to run through the full stack of driver code. To do this, an operating system call would establish a Virtual Interface channel and memory-map its buffers and queues into application address space. In addition, a library to encapsulate the low-level details of interfacing to the channel would facilitate use of such Virtual Interface connections. The library could also automatically establish redundant Virtual Interface channel pairs and manage sharing or failing over between them, without requiring any effort or awareness from the calling application.

The embodiments described above emulated Ethernet internally over an ATM-like fabric. The design may be changed to use an internal Ethernet fabric which would simplify much of the architecture, e.g., obviating the need for emulation features. If the external network communicates according to ATM, another variation would use ATM internally without emulation of Ethernet and the ATM could be communicated externally to the external network when so addressed. Another variation would allow ATM internally to the platform (i.e., without emulation of Ethernet) and only external communications are transformed to Ethernet. This would streamline internal communications but require emulation logic at the controller.

Certain embodiments deploy PANs based on software configuration commands. It will be appreciated that deployment may be based on programmatic control. For example, more processors may be deployed under software control during peak hours of operation for that PAN, or corresponding more or less storage space for a PAN may be deployed under software algorithmic control.

It will be appreciated that the scope of the present invention is not limited to the above described embodiments, but rather is defined by the appended claims; and that these claims will encompass modifications of and improvements to what has been described.

What is claimed is:

1. A platform for automatically deploying at least one virtual processing area network, in response to software commands, said platform comprising:

a plurality of computer processors connected to an internal communication network;

29

at least one control node in communication with an external communication network and in communication with an external storage network having an external storage address space, wherein the at least one control node is connected to the internal communication network and thereby in communication with the plurality of computer processors, said at least one control node including logic to receive messages from the plurality of computer processors, wherein said received messages are addressed to the external communication network and to the external storage network and said at least one control node including logic to modify said received messages to transmit said modified messages to the external communication network and to the external storage network;

configuration logic for receiving and responding to said software commands, said software commands specifying (i) a number of processors for a virtual processing area network (ii) a virtual local area network topology defining interconnectivity and switching functionality among the specified processors of the virtual processing area network, and (iii) a virtual storage space for the virtual processing area network, said configuration logic including logic to select, under programmatic control, a corresponding set of computer processors from the plurality of computer processors, to program said corresponding set of computer processors and the internal communication network to establish the specified virtual local area network topology, and to program the at least one control node to define a virtual storage space for the virtual processing area network, said virtual storage space having a defined correspondence to a subset of the external storage address space of the external storage network; and

wherein the plurality of computer processors and the at least one control node include network emulation logic to emulate Ethernet functionality over the internal communication network.

2. The platform of claim 1 wherein the internal communication network is a point-to-point switch fabric.

3. A platform for automatically deploying at least one virtual processing area network, in response to software commands, said platform comprising:

a plurality of computer processors connected to an internal communication network;

at least one control node in communication with an external communication network and in communication with an external storage network having an external storage address space, wherein the at least one control node is connected to the internal communication network and thereby in communication with the plurality of computer processors, said at least one control node including logic to receive messages from the plurality of computer processors, wherein said received messages are addressed to the external communication network and to the external storage network and said at least one control node including logic to modify said received messages to transmit said modified messages to the external communication network and to the external storage network;

configuration logic for receiving and responding to said software commands, said software commands specifying (i) a number of processors for a virtual processing area network (ii) a virtual local area network topology defining interconnectivity and switching functionality among the specified processors of the virtual processing area network, and (iii) a virtual storage space for the

30

virtual processing area network, said configuration logic including logic to select, under programmatic control, a corresponding set of computer processors from the plurality of computer processors, to program said corresponding set of computer processors and the internal communication network to establish the specified virtual local area network topology, and to program the at least one control node to define a virtual storage space for the virtual processing area network, said virtual storage space having a defined correspondence to a subset of the external storage address space of the external storage network;

wherein the at least one control node receives, via the internal communication network, storage messages from said corresponding set of computer processors, and wherein the at least one control node includes logic to extract an address from a received storage message, to identify the defined corresponding address in the external storage address space, and to provide messages on the external storage network corresponding to the received storage messages and having the corresponding address; and

wherein the at least one control node includes logic to buffer data corresponding to write messages received from a computer processor of said corresponding set of computer processors and to provide the buffered data in the corresponding message provided to the external storage network.

4. A platform for automatically deploying at least one virtual processing area network, in response to software commands, said platform comprising:

a plurality of computer processors connected to an internal communication network; at least one control node in communication with an external communication network and in communication with an external storage network having an external storage address space, wherein the at least one control node is connected to the internal communication network and thereby in communication with the plurality of computer processors, said at least one control node including logic to receive messages from the plurality of computer processors, wherein said received messages are addressed to the external communication network and to the external storage network and said at least one control node including logic to modify said received messages to transmit said modified messages to the external communication network and to the external storage network;

configuration logic for receiving and responding to said software commands, said software commands specifying (i) a number of processors for a virtual processing area network (ii) a virtual local area network topology defining interconnectivity and switching functionality among the specified processors of the virtual processing area network, and (iii) a virtual storage space for the virtual processing area network, said configuration logic including logic to select, under programmatic control, a corresponding set of computer processors from the plurality of computer processors, to program said corresponding set of computer processors and the internal communication network to establish the specified virtual local area network topology, and to program the at least one control node to define a virtual storage space for the virtual processing area network, said virtual storage space having a defined correspondence to a subset of the external storage address space of the external storage network;

31

wherein the at least one control node receives, via the internal communication network, storage messages from said corresponding set of computer processors, and wherein the at least one control node includes logic to extract an address from a received storage message, to identify the defined corresponding address in the external storage address space, and to provide messages on the external storage network corresponding to the received storage messages and having the corresponding address; and

wherein the at least one control node receives storage messages from the external storage network, and wherein the at least one control node includes logic to identify a corresponding computer processor or control node that the received message is responsive to, and to provide a corresponding message to the identified computer processor or control node.

5. A method of automatically deploying at least one virtual processing area network, in response to software commands, said method comprising the acts of:

- providing a platform having a plurality of computer processors and at least one control node connected to an internal communication network, wherein the at least one control node is in communication with an external communication network and an external storage network having an external storage address space;
- receiving software commands specifying (i) a number of processors for a virtual processing area network, (ii) a virtual local area network topology defining interconnectivity and switching functionality among the specified processors of the virtual processing area network, and (iii) virtual storage space for the virtual processing area network;
- under programmatic control and in response to the software commands, selecting a corresponding set of computer processors for the virtual processing area network;
- under programmatic control and in response to the software commands, programming said corresponding set of computer processor; and the internal communication network to establish the specified virtual local area network topology providing communication among said corresponding set of computer processors but excluding the processors from the plurality not in said set;
- under programmatic control and in response to the software commands, programming the at least one control node to define a virtual storage space of the virtual processing network, said virtual storage space having a defined correspondence to a subset of the external storage address space of the external storage network;
- wherein messages from the plurality of computer processors to the external communication network and to the external storage network are received and modified by the least one control node which transmits the modified messages to the external communication network and to the external storage network; and
- wherein the plurality of computer processors and the at least one control node emulate Ethernet functionality over the internal communication network.

6. The method of claim 5 wherein the internal communication network is a point-to-point switch fabric and wherein the emulation of Ethernet functionality is provided over the point-to-point switch fabric.

7. A method of automatically deploying at least one virtual processing area network, in response to software commands, said method comprising the acts of:

32

providing a platform having a plurality of computer processors and at least one control node connected to an internal communication network, wherein the at least one control node is in communication with an external communication network and an external storage network having an external storage address space;

receiving software commands specifying (i) a number of processors for a virtual processing area network, (ii) a virtual local area network topology defining interconnectivity and switching functionality among the specified processors of the virtual processing area network, and (iii) virtual storage space for the virtual processing area network;

under programmatic control and in response to the software commands, selecting a corresponding set of computer processors for the virtual processing area network;

under programmatic control and in response to the software commands, programming said corresponding set of computer processor; and the internal communication network to establish the specified virtual local area network topology providing communication among said corresponding set of computer processors but excluding the processors from the plurality not in said set;

under programmatic control and in response to the software commands, programming the at least one control node to define a virtual storage space of the virtual processing network, said virtual storage space having a defined correspondence to a subset of the external storage address space of the external storage network;

wherein messages from the plurality of computer processors to the external communication network and to the external storage network are received and modified by the least one control node, which transmits the modified messages to the external communication network and to the external storage network;

wherein the at least one control node receives, via the internal communication network, storage messages from said corresponding set of computer processors, axed wherein the at least one control node extracts an address from a received storage message, identifies the defined corresponding address in the external storage address space, and provides messages on the external storage network corresponding to the received storage messages and having the corresponding address; and

wherein the at least one control node buffers data corresponding to write messages received from a computer processor of said corresponding set of computer processors and provides the buffered data in the corresponding message provided to the external storage network.

8. A method of automatically deploying at least one virtual processing area network, in response to software commands, said method comprising the acts of:

- providing a platform having a plurality of computer processors and at least one control node connected to an internal communication network, wherein the at least one control node is in communication with an external communication network and an external storage network having an external storage address space;
- receiving software commands specifying (i) a number of processors for a virtual processing area network, (ii) a virtual local area network topology defining interconnectivity and switching functionality among the speci-

33

fied processors of the virtual processing area network, and (iii) virtual storage space for the virtual processing area network;

under programmatic control and in response to the software commands, selecting a corresponding set of computer processors for the virtual processing area network;

under programmatic control and in response to the software commands, programming said corresponding set of computer processor; and the internal communication network to establish the specified virtual local area network topology providing communication among said corresponding set of computer processors but excluding the processors from the plurality not in said set;

under programmatic control and in response to the software commands, programming the at least one control node to define a virtual storage space of the virtual processing network, said virtual storage space having a defined correspondence to a subset of the external storage address space of the external storage network; wherein messages from the plurality of computer processors to the external communication network and to the

34

external storage network are received and modified by the least one control node, which transmits the modified messages to the external communication network and to the external storage network;

wherein the at least one control node receives, via the internal communication network, storage messages from said corresponding set of computer processors, axed wherein the at least one control node extracts an address from a received storage message, identifies the defined corresponding address in the external storage address space, and provides messages on the external storage network corresponding to the received storage messages and having the corresponding address; and

wherein the at least one control node receives storage messages from the external storage network, and wherein the at least one control node identifies a corresponding computer processor or control node that the received message is responsive to, and provides a corresponding message to the identified computer processor or control node.

* * * * *

UNITED STATES PATENT AND TRADEMARK OFFICE
CERTIFICATE OF CORRECTION

PATENT NO. : 7,231,430 B2
APPLICATION NO. : 10/038353
DATED : June 12, 2007
INVENTOR(S) : Vern Brownell et al.

Page 1 of 1

It is certified that error appears in the above-identified patent and that said Letters Patent is hereby corrected as shown below:

In the specification

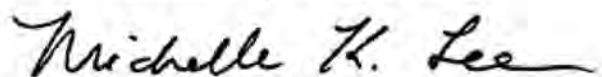
In Column 23, line 23, after "processor-side" insert --storage--
line 40, before "includes" delete "logic 720, Logic 720" insert --storage logic 620,
processor side storage logic 620--

In Column 24, line 43, after "link" delete "725" insert --625--
line 44, after "logic" delete "720" insert --620--
line 45, after "logic" delete "715" insert --615--
line 48, after "node-side" delete "logic 715" insert --storage logic 615--
line 53, after "processor-side" delete "logic 720" insert --storage logic 620--
line 54, after "logic", delete "715" insert --615--
line 64, after "processor-side" delete "logic 720" insert --storage logic 620--

In Column 25, line 4, after "logic" delete "715" insert --615--
line 5, after "logic" delete "715" insert --615--
line 14, after "logic" delete "715" insert --615--
line 17, before "communications" delete "logic 715" insert --storage logic 615--
line 23, after "logic" delete "715" insert --615--
line 32, after "processor-side" delete "logic 720" insert --storage logic 620--
line 52, after "node-side" delete "logic 715" insert --storage logic 615--
line 61, after "node-side" delete "logic 715" insert --storage 615--
line 64, after "node-side" delete "logic 715" insert --storage logic 615--

In Column 26, line 10, after "node-side" delete "logic 715" insert --storage logic 615--
line 11, after "node-side" delete "logic 715" insert --storage logic 615--
line 29, after "side" delete "logic 715" insert --storage logic 615--
line 58, after "node-side" delete "logic 715" insert --storage logic 615--

Signed and Sealed this
Seventh Day of June, 2016



Michelle K. Lee
Director of the United States Patent and Trademark Office

CERTIFICATE OF COMPLIANCE

Pursuant to Fed. R. App. P. 27(d) and 32(g), the undersigned hereby certifies that this Motion complies with the type-volume limitation of Circuit Rule 27(d).

1. Exclusive of the accompanying documents as authorized by Fed. R. App. P. 27(a)(2)(B) and the exempted portions of the motion as provided by Fed. R. App. P. 27(d)(2) and 32(f), the Brief contains 12,827 words.

2. The Motion has been prepared in a proportionally spaced typeface using Microsoft Word in 14-point Times New Roman font as provided by Fed. R. App. P. 32(a)(5)-(6). As permitted by Fed. R. App. P. 32(g), the undersigned has relied upon the word count feature of this word processing system in preparing this certificate.

Date: April 26, 2023

By: /s/ Matthew C. Holohan
Robert R. Brunelli
rbrunelli@sheridanross.com
Matthew C. Holohan
mholohan@sheridanross.com
SHERIDAN ROSS P.C.
1560 Broadway, Suite 1200
Denver, Colorado 80202
Telephone: (303) 863-9700
Facsimile: (303) 863-0223
litigation@sheridanross.com